

PROCEEDINGS

OF THE SOCIETY OF PHOTO-OPTICAL INSTRUMENTATION ENGINEERS

VOLUME 66

Efficient Transmission of
Pictorial Information



August 21-22, 1975 • San Diego, California

*Particularly
See pg 90*

Proceedings of the Society of Photo-Optical Instrumentation Engineers

VOLUME 66

Efficient Transmission of
Pictorial Information

August 21-22, 1975 • San Diego, California

Andrew G. Tescher
Editor

Presented by
The Society of Photo-Optical Instrumentation Engineers

In cooperation with
Department of the Army, United States Army Electronics Command, Night Vision Laboratory
Department of the Navy, Office of Naval Research
Energy Research Development Agency
Environmental Research Institute of Michigan
Institute of Electrical & Electronic Engineers, San Diego Section
The Institute of Optics, University of Rochester
Jet Propulsion Laboratory
National Aeronautics & Space Administration, Ames Research Center
National Aeronautics & Space Administration, Headquarters, Washington, D. C.
National Science Foundation
The Optical Society of America
U. S. Department of Commerce, National Oceanic and Atmospheric Administration

Copyright © 1975 by the Society of Photo-Optical Instrumentation Engineers, 338 Tejon Place, Palos Verdes Estates, California 90274 U.S.A. All rights reserved. This book or any part thereof must not be reproduced in any form without the written permission of the publisher. Printed in U.S.A.

Contents

Author Index	v
Seminar Committee	vii
Reasons for Data Compression—An Introduction.	ix
Andrew G. Tescher, The Aerospace Corporation	
 SESSION I. FUNDAMENTALS OF IMAGE DATA COMPRESSION	1
An Overview of Human Observer Characteristics and their Effect on Image Transmission and Display	2
Thomas G. Stockham, Jr., University of Utah	
Coding of Two-Tone Images	5
T. S. Huang, Purdue University	
Source Coding of Television Signals Using Interframe Techniques.	9
B. G. Haskell and P. L. Gordon, Bell Telephone Laboratories, Inc.	
Bandwidth Compression of Multispectral Data	23
A. Habibi and A. S. Samulon, TRW Systems Group	
Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices	36
H. J. Whitehouse, R. W. Means, E. H. Wrench, Naval Undersea Center	
Technology of Charge-Coupled Devices for Video Bandwidth Reduction	48
D. D. Buss, R. W. Brodersen, C. R. Hewes, A. F. Tasch, Jr., Texas Instruments, Inc.	
 SESSION II. ADVANCED IMPLEMENTATIONS	57
Real-Time Video Compression Algorithm for Hadamard Transform Processing	58
Scott C. Knauer, Ames Research Center, NASA	
An Advanced Imaging Communication System for Planetary Exploration	70
Robert F. Rice, Jet Propulsion Laboratory	
> Advanced Facsimile System and Network Applications.	90
John W. Watkins, Harris Corporation, Electronic Systems Division	
An Operational Video Data Compression System for ATS and ITOS	94
R. L. Kutz, NASA/Goddard Space Flight Center; L. D. Davisson, Univ. of Southern California	
A CAQ Bandwidth Reduction System for RPV Video Transmission	101
James J. Pearson, Lockheed Palo Alto Research Laboratory	
Intraframe and Interframe Adaptive Transform Coding.	108
Clifford Reader, Aeronutronic Ford Corporation	
 SESSION III. Panel: IMPACTS OF IMAGE DATA COMPRESSION	119
 SESSION IV. POTENTIAL IMPLEMENTATIONS	121
Joint Pattern Recognition Data Compression Concept for ERTS Multispectral Imaging.	122
Edward E. Hilbert, Jet Propulsion Laboratory	
A Dual Mode Nonlinear Interpolative Compressor for SAR Images	138
Wen-hsiung Chen, Aeronutronic Ford Corporation	
A Fast Two-Dimensional Karhunen-Loeve Transform.	144
Robert M. Haralick, Norman Griswold, Nimitra Kattiyakulwanich, University of Kansas Center for Research, Inc., Remote Sensing Laboratory	
Singular Value Decomposition for Image Coding.	160
Harry C. Andrews, University of Southern California, Image Processing Institute	
TV Bandwidth Compression	161
Paul Wintz, Purdue University; Jim Wilson, WINTEK Corp.; Ernie Schmidt, Aeronutronic Ford Corp.	
DPCM Quantization Error Reduction for Image Coding	167
William K. Pratt, Michael N. Huhns, University of Southern California	
Combined Spatial and Temporal Coding of Digital Image Sequences	172
John A. Roese, Naval Undersea Center; Guner S. Robinson, University of Southern California	

⇒

SESSION V. RELATED TOPICS	181
Laser Scanner/Recorders for Image Transmission and Computer Processing	182
W. F. Schreiber, U. F. Gronemann, D. E. Troxel, Massachusetts Institute of Technology	
Real-Time Image Data Acquisition/Retrieval (RIDAR) System.	190
Harold H. Kantner, CAI, a Division of Bourns, Inc.	
An Investigation of MSE Contributions in Transform Image Coding Schemes	196
John R. Parsons, Andrew G. Tescher, The Aerospace Corporation	
A Comparison of Hardware Implementations of the Hadamard Transform for Real-Time Image Coding. . .	207
Stephen C. Noble, Air Force Avionics Laboratory, Wright Patterson Air Force Base	
A Teaching Stereo-Video Microscope	212
James F. Butterfield, Dimensional Television Corporation	
Pictorial Information Transmission through Simulation	215
Robert T. P. Wang, Honeywell, Marine Systems Division California Center	
Three-Dimensional Perception Produced by Two 2-Dimensional Geometric Patterns Presented with	
Brief Exposure Times and Time Intervals.	223
Kiyoe Mizusawa, The Pennsylvania State University	

Author Index

Andrews, Harry C. (University of Southern California, Image Processing Institute).	160
<i>Singular Value Decomposition for Image Coding</i>	
Brodersen, R. W. (Texas Instruments, Inc.).	48
<i>Technology of Charge-Coupled Devices for Video Bandwidth Reduction</i>	
Buss, D. D. (Texas Instruments, Inc.).	48
<i>Technology of Charge-Coupled Devices for Video Bandwidth Reduction</i>	
Butterfield, James F. (Dimensional Television Corporation).	212
<i>A Teaching Stereo-Video Microscope</i>	
Chen, Wen-hsiung (Aeronutronic Ford Corporation).	138
<i>A Dual Mode Nonlinear Interpolative Compressor for SAR Images</i>	
Davissou, L. D. (University of Southern California).	94
<i>An Operational Video Data Compression System for ATS and ITOS</i>	
Gordon, P. L. (Bell Telephone Laboratories, Inc.).	9
<i>Source Coding of Television Signals Using Interframe Techniques</i>	
Griswold, Norman (University of Kansas Center for Research, Inc.).	144
<i>A Fast Two-Dimensional Karhunen-Loeve Transform</i>	
Gronemann, U. F. (Massachusetts Institute of Technology).	182
<i>Laser Scanner/Recorders for Image Transmission and Computer Processing</i>	
Habibi, A. (TRW Systems Group).	23
<i>Bandwidth Compression of Multispectral Data</i>	
Haralick, Robert M. (University of Kansas Center for Research, Inc.).	144
<i>A Fast Two-Dimensional Karhunen-Loeve Transform</i>	
Haskell, B. G. (Bell Telephone Laboratories, Inc.).	9
<i>Source Coding of Television Signals Using Interframe Techniques</i>	
Hewes, C. R. (Texas Instruments, Inc.).	48
<i>Technology of Charge-Coupled Devices for Video Bandwidth Reduction</i>	
Hilbert, Edward E. (Jet Propulsion Laboratory).	122
<i>Joint Pattern Recognition/Data Compression Concept for ERTS Multispectral Imaging</i>	
Huang, T. S. (Purdue University).	5
<i>Coding of Two-Tone Images</i>	
Huhns, Michael N. (University of Southern California).	167
<i>DPCM Quantization Error Reduction for Image Coding</i>	
Kantner, Harold H. (CAI, a Division of Bourns, Inc.).	190
<i>Real-Time Image Data Acquisition/Retrieval (RIDAR) System</i>	
Kattiyakulwanich, Nimitra (University of Kansas Center for Research, Inc.).	144
<i>A Fast Two-Dimensional Karhunen-Loeve Transform</i>	
Knauer, Scott C. (Ames Research Center/NASA).	58
<i>Real-Time Video Compression Algorithm for Hadamard Transform Processing</i>	
Kutz, R. L. (NASA/Goddard Space Flight Center).	94
<i>An Operational Video Data Compression System for ATS and ITOS</i>	
Means, R. W. (Naval Undersea Center).	36
<i>Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices</i>	
Mizusawa, Kiyoe (The Pennsylvania State University).	223
<i>Three-Dimensional Perception Produced by Two 2-Dimensional Geometric Patterns Presented with Brief Exposure Times and Time Intervals</i>	
Noble, Stephen C. (Air Force Avionics Laboratory, Wright-Patterson Air Force Base).	207
<i>A Comparison of Hardware Implementations of the Hadamard Transform for Real-Time Image Coding</i>	
Parsons, John R. (The Aerospace Corporation).	196
<i>An Investigation of MSE Contributions in Transform Image Coding Schemes</i>	
Pearson, James J. (Lockheed Palo Alto Research Laboratory).	101
<i>A CAO Bandwidth Reduction System for RPV Video Transmission</i>	

⇒

Pratt, William K. (University of Southern California)	167
<i>DPCM Quantization Error Reduction for Image Coding</i>	
Reader, Clifford (Aeronutronic Ford Corporation)	108
<i>Intraframe and Interframe Adaptive Transform Coding</i>	
Rice, Robert F. (Jet Propulsion Laboratory).	70
<i>An Advanced Imaging Communication System for Planetary Exploration</i>	
Robinson, Guner, S. (University of Southern California).	172
<i>Combined Spatial and Temporal Coding of Digital Image Sequences</i>	
Roesse, John A. (Naval Undersea Center)	172
<i>Combined Spatial and Temporal Coding of Digital Image Sequences</i>	
Samulon, A. S. (TRW Systems Group).	23
<i>Bandwidth Compression of Multispectral Data</i>	
Schmidt, Ernie (Aeronutronic Ford Corporation)	161
<i>TV Bandwidth Compression</i>	
Schreiber, W. F. (Massachusetts Institute of Technology)	182
<i>Laser Scanner/Recorders for Image Transmission and Computer Processing</i>	
Stockham, Jr., Thomas G. (University of Utah).	2
<i>Overview of Human Observer Characteristics & their Effect on Image Transmission & Display</i>	
Tasch, Jr., A. F. (Texas Instruments, Inc.)	48
<i>Technology of Charge-Coupled Devices for Video Bandwidth Reduction</i>	
Teschner, Andrew G. (The Aerospace Corporation).	ix, 196
<i>Reasons for Data Compression—An Introduction</i>	
<i>An Investigation of MSE Contributions in Transform Image Coding Schemes</i>	
Troxel, D. E. (Massachusetts Institute of Technology)	182
<i>Laser Scanner/Recorders for Image Transmission and Computer Processing</i>	
Wang, Robert T. P. (Honeywell, Marine Systems Division California Center).	215
<i>Pictorial Information Transmission through Simulation</i>	
Watkins, John W. (Harris Corporation, Electronic Systems Division)	90
<i>Advanced Facsimile System and Network Applications</i>	
Whitehouse, H. J. (Naval Undersea Center).	36
<i>Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices</i>	
Wilson, Jim (WINTEK Corp.).	161
<i>TV Bandwidth Compression</i>	
Wintz, Paul (Purdue University)	161
<i>TV Bandwidth Compression</i>	
Wrench, E. H. (Naval Undersea Center)	36
<i>Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices</i>	

Seminar Committee

General Chairman

Andrew G. Tescher, The Aerospace Corporation

Technical Program Committee

Harry C. Andrews, University of Southern California

Ali Habibi, TRW Systems Group

John R. Parsons, The Aerospace Corporation and University of Arizona

Session I. Fundamentals of Image Data Compression

Chairman, Andrew G. Tescher, The Aerospace Corporation

Session II. Advanced Implementations

Chairman, Ali Habibi, TRW Systems Group

Session III. Impacts of Image Data Compression

Panel Moderator, Andrew G. Tescher, The Aerospace Corporation

Session IV. Potential Implementations

Chairman, J. R. Parsons, The Aerospace Corporation

Session V. Related Topics

Chairman, Harry C. Andrews, Image Processing Institute, University of Southern California

Reasons for Data Compression—An Introduction

Andrew G. Tescher

The Aerospace Corporation, P. O. Box 92957, Los Angeles, California 90009

Visual data is the primary form by which man acquires information. However, the required information and its potential user may be separated by large distances. Although conventional communication channels are available, the large amount of information present in pictorial data quite often taxes this available communication bandwidth. At the practical level, pictorial data transmission may be prohibitively expensive.

The seminar on "Efficient Transmission of Pictorial Information" attempted to cover various problem areas associated with visual information transfer technology. The twenty-six, primarily invited, papers address current and future problem areas. The various subject areas can be grouped into image representation, image coding, psychophysical considerations, channel effects, image quality, data management, and display technology. There is a balanced ratio of primarily theoretical discussions and those papers which emphasized hardware development.

By design, the presentation level of the various papers is diverse. On one side of the spectrum, the several review papers, whose authors are well-respected experts of long experience in this field, are primarily provided for the newcomers to the subject of image compression. The majority of the papers, however, present new and valuable results and in most cases advance the state-of-the-art of their respective specializations.

In addition to the formal presentations, an informal panel discussion was organized and moderated by this author. The panelists were F. Billingsley of JPL, H. M. Federhen of ARPA, C. R. Hewes of Texas Instruments, Inc., and P. A. Wintz of Purdue University. The panel, interactively with the numerous individuals offering comments and suggestions, has reiterated the need for new solutions in the field of pictorial data communication.

Regarding just how fast new solutions will come, the opinion is diverse. Perhaps based on a personal bias leaning towards optimism, this author believes the new and effective solutions will come soon. This belief is supported by the extensive and high quality activity in the field of image compression and related technologies.

Andrew G. Tescher, Editor

Session I
FUNDAMENTALS OF IMAGE DATA COMPRESSION

Chairman
ANDREW G. TESCHER
The Aerospace Corporation

AN OVERVIEW OF HUMAN OBSERVER CHARACTERISTICS AND THEIR EFFECT
ON IMAGE TRANSMISSION AND DISPLAY

Thomas G. Stockham, Jr.
Departments of Computer Science and Electrical Engineering
University of Utah
Salt Lake City, Utah 84112

The science and technology of image transmission and display has evolved primarily from the disciplines of computer science, electrical engineering and physics. Thus, it is only natural that techniques and attitudes which have developed are characterized by the style in which people in these areas approach the subject of images and their transmission, display, and processing. For example, one finds many important differences between the methods of television and those of photography. Moreover, an even greater contrast is found when comparing the methods of electrical engineering, physics, television and photography with those now understood to be employed by the human eye and the human visual system. Specifically, let us contrast the method by which television, photography and the human visual system, represent image information. In television (particularly digital imaging), image values are represented by signals analogous to quantities of light. They are called intensities. In photography, on the other hand, the representing quantities are concentrations of silver or dyes. Consequently, due to the natural exponentiating laws governing the interaction of light with these media, they are analogous to the logarithm of quantities of light. They are called densities. The human visual system, on the other hand, (while being generally logarithmically sensitive) moves a large step further away from representation by physical quantities of light and produces at very early stages in its processing highly modified versions of the patterns of light or their logarithms. The natural question then arises, why should the human visual system try to do this; and since it does, what consequences are implied in terms of the television and photographic presentations normally employed?

The answers to these questions rest crucially on the issue of errors. In any transmission or display system, errors will be committed. These are unavoidable and are a result of the physical limitations encountered. In a broad variety of applications, the most important forms of error encountered are those imposed by limited dynamic range and various forms of noise. The classic dilemma one faces in the disciplines of image transmission and display is that a compromise must be effective between the conflicting constraints of dynamic range and noise. On the one hand, one wishes to make signals larger so that the noise may be rendered negligible. At the same time, large signals are precluded by the limitation in dynamic range in the form of distortions which effect the signals when their values are made too variable. More specifically, in any transmission and/or display design, the natural goal would seem to be the attainment of fidelity reproduction. Unfortunately, the demands of typical imagery upon transmission and display systems is usually so severe that this goal cannot be reasonably met. The distortions due to limited dynamic range and noise, especially the former, are relentlessly unavoidable and so the designer must content himself with one form of distortion or another.

Fortunately, the human visual system itself produces a large quantity of distortion. This phenomenon may be exploited by trading off undesirable forms for forms which will be encountered naturally anyway. The distortions produced by the human visual system are often referred to as optical illusions. The simplest of these is the gray scale illusion which reveals the logarithmic sensitivity of the human visual system alluded to above⁽¹⁾. This illusion is responsible for the fact that we do not turn the lights on during the day although they add just as much light then as at night. Another ramification is that gray scales must be arranged in exponential progression to appear arithmetic to the observer.

Other illusions which are spatial in character are much more important and striking, however. The simplest of these is the illusion of simultaneous contrast which permits two neutral gray shades to appear so different from one another at the same time, that one can assign the names near-white to one and near-black to the other. Careful study of these illusions has permitted researchers to formulate signal processing models for the early processing stages of the human visual system. These models (e.g. consider those of Stockham⁽²⁾, Baudelaire⁽³⁾, Frei⁽⁴⁾, and Baxter⁽⁵⁾) predict the human visual illusions or distortions well enough to permit their use in defining useful objective measures of how different two images (one original, the other distorted) will look from one another.

Applications of these models to problems in image transmission and display have already yielded significant advantages. For example, when the model is used to evaluate the distortions produced by a given display instrument, it is often possible to predict how the physical performance of that display may be significantly relaxed without

producing appreciable visual degradation. Additionally, by using the visual model as a processor prototype, it is possible to "look at the image" before it is displayed or transmitted by first processing the image with the visual model processor and then transmitting or displaying it. This procedure places the image information in a signal space maximally compatible with the most recent knowledge of the human visual system before the degradations characteristic of either transmission or display are allowed to detract from the quality.

The advantages are significant as can be seen from (2) and (6). Rom shows how in the case of transmission with coding, it is possible by including human observer characteristics in the criterion for error to produce significantly superior subjective image transmission. First, the image is processed through the visual model. The resulting signal is then coded and transmitted. The received image is then passed through the inverse of the visual model and displayed. Subject to the validity of the model, the effect is as if the distortions due to coding were not allowed to enter the system until the image had been processed significantly by the human visual system itself.

The most important problem in image display is that associated with limited dynamic range. Cathode ray tubes, photographic transparencies, reflection photographs (i.e. prints), and images printed with ink on paper all face the basic light-to-dark ratio limitations of their media. For example, a good photographic film is not capable of reflecting much more than 90% of the light that strikes it. Neither is it capable of reflecting much less than 1% of the light that strikes it. The result is a dynamic range limitation of about 100:1. Unfortunately, high fidelity images such as those seen by the human eye in the real world, obtain dynamic ranges far in excess of 1000:1 or even 10,000:1. This limitation can and has been overcome throughout the years by various techniques. Now, however, knowledge of the human visual system allows us to effect the same dynamic range limiting processes on the raw images to be displayed that the eye uses naturally. Since these processes take place anyway when the display is viewed, little is lost by also invoking them before the limitations of the display or transmission medium is encountered. In addition, by the very nature of the processes which the visual system uses, double processing is not subjectively disagreeable and often can hardly be distinguished from the single natural process which takes place in direct viewing of the original scene. Thus, by deliberately distorting the image before transmission and display by using visual model processing, one can avoid the much more disagreeable distortions which would be otherwise encountered due to the transmission or display.

As our knowledge of human observer characteristics grows, it is possible to formulate more sophisticated and complex visual models, and our abilities to cope with the problems being described here increases. While the latest work in visual modeling has included such aspects of vision as color constancy and contrast⁽⁷⁾⁽⁴⁾ as well as the relatively recently disclosed frequency channel model of vision⁽⁸⁾⁽⁵⁾, it is anticipated that in the future, much more sophisticated aspects will also be embraced. Such aspects will undoubtedly include edge detection, motion sensitivity, non-uniform visual acuity over the field and perhaps higher forms of image abstraction.

Studies are already underway in an effort to discover methods for analyzing images by synthesis. One possible synthesis method is through the modern technology of shaded image synthesis by computer. While image transmission systems employing coding have for the most part made use of signal properties to achieve their bandwidth reduction, it is also possible to achieve potentially much greater reduction by coding to a level embracing the geometric structure constituting the scene which has been imaged. Such a breakthrough would be based, not only on the structure of images themselves, but also upon the way in which that structure is perceived by the visual system. An analogy with speech bandwidth compression, which is capable of much greater information rate savings than the best general sound bandwidth compression methods, is an informative one. The basic structure of speech and the mechanism which produces it, is exploited in such bandwidth compression methods. Techniques for detecting the parameters of vocal chord motion and the shape of the human vocal tract directly from the acoustic signal are at the heart of such techniques. In addition, the nature of the human hearing is exploited in effecting compromises which must be introduced during the analysis.

So it is that one might expect the most modern efforts in image transmission, display and processing to be strongly influenced by a knowledge of the human observer characteristics. The implication of this statement is that the image processing scientist must be well versed not only in the physics of optics, the electrophysics of sensing, the electronics and chemistry of transmission, and the analog and digital disciplines of processing, but also in the psychophysics, biophysics and psychology of vision.

References

1. Cornsweet, T.N. *Visual Perception*. Chapter X. New York:Academic Press, 1970, pp. 249-253.

2. Stockham, T.G., Jr. "Image Processing in the Context of a Visual Model," Proc. IEEE, vol. 60, pp. 828-842, July 1972.
3. Colas-Baudelaire, Patrick. "Digital Picture Processing and Psychophysics: A Study of Brightness Perception," Computer Science Dept., University of Utah, Salt Lake City, Utah, UTEC-CSc-74-025, March 1973.
4. Frei, Werner. "A New Model of Color Vision and Some Practical Implications," Univ. of Southern California Image Processing Institute, Semiannual Technical Report, March 31, 1974, USCIP Report 530, pp. 128-143.
5. Baxter, Brent S. "Image Processing in the Human Visual System," Computer Science Dept., University of Utah, Salt Lake City, Utah, (in publication).
6. Rom, Raphael. "Image Transmission and Coding Based on Human Vision," Computer Science Dept., University of Utah, Salt Lake City, Utah, (in publication).
7. Cornsweet, T.N. *Visual Perception*. Chapter XIII. New York:Academic Press, 1970.
8. Campbell, F.W., Cooper, F.F. and Christina Enrath-Cndgell. "The Spacial Cells of the Cat." Journal of Physiology 203, p. 223.

CODING OF TWO-TONE IMAGES

T. S. Huang
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

Abstract

We give a brief overview of efficient coding methods for two-tone images, especially: white block skipping and runlength coding.

I. Introduction

In this tutorial paper we discuss some aspects of the digitization and efficient coding of two-tone images, such as business documents, line drawings, and weather maps.

After a brief discussion on sampling and quantization of nominally two-tone images, we describe the two main classes of efficient coding methods: white skipping, and contour coding. Perhaps the most commonly used efficient coding scheme is runlength coding. The remaining part of the paper concentrates on easily implementable codes for runlengths.

II. Digitization

We shall concern ourselves with the digital transmission of images. Then, the first step is to sample the image and quantize the amplitude of each sample. For two-tone images we quantize the amplitude to two levels⁽¹⁾; thus each sample is represented by one bit. The number of samples required depends on the image at hand. We take two examples: business letters and newspaper pages.

In analog transmission of business letters, a resolution of around 100 lines per inch gives satisfactory reproductions. For digital transmission where the amplitude of each sample is quantized to only two levels, the same resolution of 100 lines per inch yields reproductions which are marginally acceptable, the main defect being the staircase appearance of the boundaries between the black and the white. We can improve the image quality by either increasing the resolution or using more than two levels for each sample⁽¹⁾. Keeping two levels per sample, a linear resolution of around 150 points per inch (ppi) is required for satisfactory image quality. At 150 ppi and 1 bit per sample, a 8 1/2" x 11" letter gives us approximately 2 million bits. To transmit such a letter using PCM through a telephone channel with a 2400 bits/sec modem takes about 14 minutes. In order to reduce the transmission time to say 1 minute, a data reduction or compression ratio of 14:1 is required.

There has been considerable interest in the digital transmission of newspaper pages⁽²⁾. It has been found that, for text material, a sampling density of 400 ppi is sufficient to ensure satisfactory reproduction quality. For halftone pictures⁽³⁾, the required sampling density depends on two factors: (1) how many equivalent brightness levels (i.e. how many different halftone dot sizes) would be desirable; and (2) how dense we have to sample in order to avoid the appearance of moire patterns.

No systematic subjective experiments have been conducted with respect to the first question. However, to extrapolate from experiments with continuous-tone pictures⁽⁴⁾, it is estimated that 64 levels are sufficient and 32 levels may be satisfactory in many cases. Let f_h and f_s be the halftone screen density and the sampling density, respectively, in ppi. If f_s is k times f_h , then we have k^2 samples per dot, which allows us to create k^2 dot sizes (by making some of the samples black). For example, for 64 levels, we need $f_s = 8f_h$; for 32 levels, $f_s = 6f_h$.

With respect to the second question, less is known. It is generally assumed that f_s must be at least $10f_h$ in order to avoid moire patterns. However, recent experiments⁽⁵⁾ showed that the factor 10 might be too conservative since no moire patterns were observed in any experiment with $f_s \geq 8f_h$.

From the above discussion, then, it seems that $f_s = 8f_h$ is a good choice for the sampling density in the case of halftone pictures. A typical value for f_h is 85 ppi. The corresponding sampling density is $f_s = f_h = 680$ ppi, which is 70% larger than what f_h is needed for the text material. Since, operationally, it is much simpler to use the same sampling density for both the text material and the halftone pictures, we will be wasting a lot of samples in the text material. In any case, it should be obvious that a huge amount of bits is required to represent newspaper pages. Even with a wideband channel, some data compression is desirable.

III. Basic Principles of Efficient Coding

To reduce transmission time, we can either do source coding to reduce data redundancy or to design higher speed modems. In this paper we are concerned with the first approach. We shall also limit ourselves to error-free coding - the receiver can reconstruct an exact copy of the original digitized image (assuming the channel is noiseless).

There are two basic principles of efficient coding for two-tone image: skipping white, and contour coding.

In most two-tone images, such as business documents and weather maps, the information is contained in black and there is a large amount of white background. We should aim to skip the white space and not transmit it.

Furthermore, to transmit the (sampled) black areas, we have to transmit only the boundary points. The receiver can fill in the interiors by black.

In sections IV and V, we discuss some practical implementations of these basic principles.

IV. White Block Skipping

Most documents and line drawings contain a large amount of white space. One approach to their efficient coding is to skip the white space. A simple way of doing that was suggested by de Coulon and Kunt⁽⁵⁾, Horlander⁽⁶⁾, and others. Each sampled scan line is divided into N-picture-element (pel) blocks. For a block containing all white pels, the 1-bit code word "0" is used. For a block containing at least one black pel, we use an (N+1)-bit code word, the first bit being "1" and the next N bits being the binary pattern of the N-pel block in question. For example, if N = 4 and a block contains two white pels followed by two black pels, then the code word for that block is "10011" - we have used "0" to represent white and "1" black. We call this code the white block skipping code.

Once a value for N is fixed, the optimum code for the 2^N possible N-pel patterns is of course the Huffman code based on the probability distribution of these patterns. However, the implementation of a Huffman code is complicated. If the image being coded contains a large amount of white, then the white block skipping (WBS) code may perform almost as well as the Huffman code and in the meantime is much simpler to implement.

For a given image or class of images, the average bit rate b_N (bits/pel) of the WBS code depends on the block size N. We would like of course to use an N which minimizes the bit rate b_N .

The WBS code does not take full advantage of the fact that most texts contain a large number of all-white scan lines.⁽⁷⁾ To take advantage of that, we propose a modified WBS code where all-white scan lines are treated specially. We use the 1-bit code word "0" to represent an all-white scan line. For a scan line containing at least one black pel, we transmit "1" followed by the regular WBS code words for that line. Assuming N = 3, then for a scan line containing the pels: WWWBBWBWBW..., we will transmit: 1011101101....". The first bit "1" indicates the scan line is not all white, and the second bit "0" represents WWW, etc. The optimum N is chosen by looking at the statistics of the image after we have discarded the all-white scan lines. Thus, by adding a one-bit overhead for each line, we have eliminated the necessity of coding and transmitting the all-white scan lines. This overhead is low on a per pel basis. For example, if we have 1,000 pels in each scan line, then the overhead is only .001 bit/pel which is negligible in most cases.

For images such as weather maps which contain very few or no all-white scan lines, the modified code reduces essentially to the regular WBS codes.

For some experimental results, see Table I. The pictures used are the same as in Ref. 5.

V. Runlength Coding and Extensions⁽⁸⁾

In runlength coding, the image is scanned linearly in the usual manner; however, instead of transmitting the individual samples, the lengths of the black and the white runs are transmitted. To achieve compression, the statistics of the runs are utilized. Some results are shown in Table I. Huffman coding was used for the runs. For a given set of probabilities for the runlengths, the most efficient code is of course the Huffman code. However, the implementation of a Huffman code requires a large dictionary and is complicated.

There are suboptimum variable-length codes which are almost as efficient as Huffman codes yet much simpler to implement. Two classes of such codes, the linear codes and the logarithmic codes are particularly suitable for graphical data⁽⁹⁾. The linear codes, which includes the Laemmle codes and the Golomb codes, has the property that the code word length increases approximately linearly with runlength. They are nearly optimum for exponential distributions. The logarithmic codes, which includes the Hasler codes⁽¹⁰⁾, are so named because the codeword length increases approximately logarithmically with runlength. They are nearly optimum for negative-power distributions.

Experimental evidence indicates that the frequency distributions of runlengths of alphanumeric data (typewritten letters, printed texts, etc.) are roughly negative-power, and that those of runlengths of line drawings and weather maps are roughly exponential. One would therefore expect that linear codes are nearly optimum for line drawings and weather maps and that logarithmic codes are nearly optimum for alphanumeric data. Such indeed turns out to be the case.

Both linear and logarithmic codes are easy to implement. The main part of the encoder or decoder is a counter.

An extension of runlength coding, called PDQ, was described in Ref. 8. In this method, essentially the changes in the black-to-white and white-to-black transition locations from scanline to scanline are coded.

Assuming that the black/white boundaries have no preferred direction, we can show that the probability distribution of the amount of change of a transition location from line to line is Cauchy, which is approximately negative-power. We therefore expect that logarithmic codes are suitable in this case.

VI. Other Coding Schemes

For other coding schemes, see Refs. 11-14. For channel error consideration, see Ref. 15.

Acknowledgement

This work was supported in part by the Advanced Research Projects Agency of the Department of Defense under Contract No. F 30602-75-C-0150.

Table I. Bit Rates of Coding Schemes

Picture Number	Bit rate (Bits/pel)		
	WBS Code	Modified WBS Code	Runlength Coding
A1 (typewritten)	.15	.10	.10
A2 (typewritten)	.30	.18	.22
A3 (circuit diagram)	.25	.25	.13
A4 (handwritten)	.19	.14	.12
A5 (weather map)	.40	.41	.24
A6 (weather map)	.37	.37	.23

References

1. Huang, T.S. and Koller, H. U., "Coding of Multilevel Images," IEEE Trans. on Communications, June 1975.
2. ANPA Research Institute Bulletin 949, All-digital page facsimile tests at the ANPA/RI Research Center, March 18, 1968.
3. Huang, T.S., "Digital Transmission of Halftone Pictures," Computer Graphics and Image Processing, September 1974.
4. Huang, T. S., "PCM Picture Transmission," IEEE Spectrum, December 1965.
5. Coulon, F. de and Kunt, M, "An Alternative to Run-length Coding for Black-and-White Facsimile," Proc. of the 1974 International Zürich Seminar on Digital Communications, Zürich, Switzerland, March 1974.
6. Horlander, F. J., IBM Disclosure, 1972.
7. Huang, T. S. and Hussain, A. B. S., "Facsimile Coding by Skipping White," to appear in IEEE Trans. on Communications.

8. Huang, T. S., "Runlength Coding and its Extensions," in Picture Bandwidth Compression, ed. by T. S. Huang and O. J. Tretiak, Gordon and Breach, 1972.
9. Huang, T. S., "Easily Implementable Suboptimum Runlength Codes," 1975 Int'l Communication Conference Record, June 16-18, 1975, San Francisco.
10. Meyr, H, Rosdolsky, H and Huang, T. S., "Optimum Runlength Codes," IEEE Trans. on Communications, June 1974.
11. Arps, R. B., "Bibliography on Digital Graphic Image Compression and Quality," IEEE Trans. on Inf. Theo., Jan. 1974.
12. Stern, P. A. and Heinlein, W. E., "Facsimile Coding Using Two-dimensional Runlength Prediction," Proc. 1974 Zürich Seminar on Digital Communications, Zürich, Switzerland, March 1974.
13. Ascher, R. N. and Nagy, G., "A Means for Achieving a High Degree of Compaction on Scan-digitized Printed Text," IEEE Trans. on Computers, Nov. 1974.
14. Session 7: Digital Facsimile Compression, 1975 Int'l Comm. Conf., June 16-18, 1975, San Francisco.
15. Koller, H. U., "Einfluss von Kanalfehlern auf Digitale Faksimile-Übertragungen," Ph.D. Thesis, Institut für Technische Physik, ETH-Zürich, Switzerland, 1975.

SOURCE CODING OF TELEVISION SIGNALS
USING INTERFRAME TECHNIQUES

by

B. G. Haskell and P. L. Gordon
Bell Telephone Laboratories, Incorporated
Holmdel, New Jersey

The channel capacity required for long distance digital transmission of video signals can be substantially reduced through a system using interframe coding. This is possible because of the amount of frame-to-frame redundancy which exists in a video source output. Through appropriate signal processing it is feasible to send only the changed area of each frame rather than the full frame provided that the previous frame values are retained in memories.

Since the very beginning of television, efforts have been underway to develop cheaper means for transmitting video signals.^(1,2) Recently, most of the attention has been focused on digital systems because, as is well known, noise does not accumulate in digital regenerators as it does in analog amplifiers, and, in addition, signal processing is much easier in a digital format.

Progress is being made on two fronts. First, the present high cost-per-bit of transmitting a digital data stream has generated interest in a number of methods for cost reduction which have general application and are not confined to data streams produced by a video signal source. However, it is important to remember that video bit-rates tend to be considerably higher than those required for voice or data transmission. The most promising techniques for more economical digital transmission include waveguides, optical fibers, laser light pipes, among others.

The second front on which progress is being made involves reducing the number of bits which have to be transmitted in a video communication system. The balance of this paper deals with this facet of the problem.

Bit-rate reduction is accomplished by eliminating, as much as possible, the substantial amount of redundant information which exists in a video signal as it leaves the camera. The cost of the signal processing which is required to reduce this redundancy determines its economic feasibility in a given system. The savings which accrue from lowering the transmission bit-rate must more than offset the cost of the required signal processing if redundancy reduction is to be economical.

Statistical Redundancy and Subjective Redundancy

If an information source such as a television camera produces statistically redundant data - that is, information which could just as well have been derived from past data - then a saving of transmission bit-rate can result if the redundant information is removed prior to transmission. In most cases this requires, at the transmitter, a capability for storing some of the past source output so that a decision can be made as to what is and what is not redundant in the present source output. Memory of past information is also required at the receiver so that the redundancy can be rederived and inserted back into the data stream in order to reproduce the original signal.

For example, in a television picture successive picture points (elements) along a line are very much alike, and redundancy reduction can be achieved by sending element-to-element differences instead of the elements themselves. The differences are small most of the time and large only occasionally; thus, an average bit-rate saving can be obtained by using short binary words to represent the more-probable, small differences and longer binary words to represent the infrequent, large differences. In Figure 1, single element delays are used to store the previous element so that differences can be generated at the sending end and picture elements can be reconstructed at the receiving end. In successive frames a picture element also changes very little on the average. To transmit frame-to-frame differences, the delays in Figure 1 should be increased from an element period to a frame period. The delay elements comprise the aforementioned memory required for redundancy removal. Statistical redundancy is not the only form of redundancy in a video signal. There is also considerable subjective redundancy; that is, information which is produced by the source, but which is not necessary for subjectively acceptable picture quality at the receiver. For example, it is well known that viewers are less sensitive to degradations near edges, i.e., large brightness transitions, in the picture. Also, viewers require less reproduced resolution for moving objects in a picture than for stationary objects. Thus, in nonscientific applications where exact reproduction of the video source output is not necessary as

long as the displayed picture is subjectively pleasing, a further reduction in transmission bit-rate can be achieved by removing subjective redundancy.

For example, the element-differential PCM coder of Figure 1 need not transmit large differences as accurately as small differences because of the viewer's insensitivity to distortion near large brightness transitions. Thus, prior to transmission, large element differences can be quantized more coarsely than small differences, thereby reducing the number of levels which must be transmitted. Since the original signal is distorted by using this type of quantization (companded quantization) the output of the transmitter delay in Figure 1 is not the actual previous element as produced by the camera. It is, instead, the previous element as it appears at the receiver with the subjective redundancy removed.

Videotelephone format pictures (1 MHz, 267 lines) can be transmitted with 16 level quantization of element differences (4 bits per picture element). The bit-rate can be reduced further by using multilength binary words for transmission; however, a buffer memory is then needed to transmit the resulting irregular data-rate over a constant bit-rate digital channel.

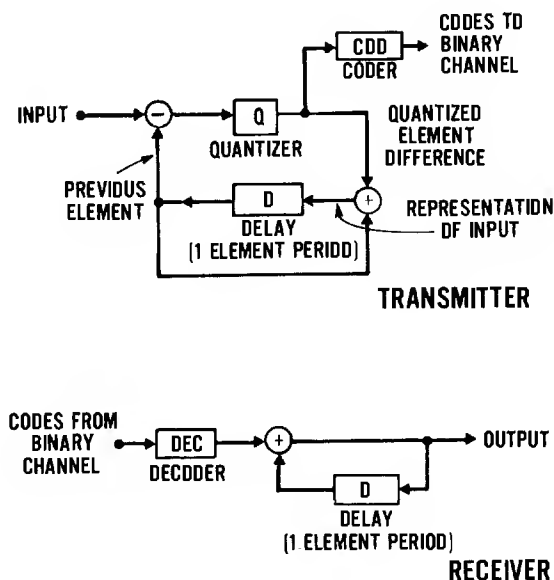


Figure 1 Transmission of element-to-element differences instead of the elements themselves. The differences are small most of the time. Thus, a bit rate reduction results if short binary code words are used for small differences and longer code words are used for large differences. Frame-to-frame differences can be transmitted if the delays are increased to a frame period.

Simple Methods of Frame-to-Frame Redundancy Reduction

In order to reduce frame-to-frame redundancy a memory or delay capable of storing an entire frame of video information is needed. At present, this requirement is the main factor determining the economic feasibility of frame-to-frame signal processing. However, it is expected that costs for digital storage will continue to decline, thereby making this type of signal processing even more attractive in the years to come. Devices for digital storage which are currently being studied include magnetic bubble memories,⁽³⁾ charge coupled devices,⁽⁴⁾ large MOS stores⁽⁵⁾ and surface acoustic wave devices.⁽⁶⁾

One method of removing frame-to-frame redundancy is simply to reduce the number of frames that are transmitted per second as shown in Figure 2. At the receiver, frames are repeated as in motion picture projection in order to avoid flicker in the display. This technique takes advantage of the fact that frame display rates must be greater than about 50 Hz to eliminate objectionable flicker, whereas something between 20 and 30 Hz is all that is required for rendition of normal motion, and less than 15 Hz for rendition of low speed movement.⁽⁷⁾

In most systems interlaced scanning already takes advantage of this phenomenon to some extent. Odd numbered lines are sent during one half frame period (field 1) and even numbered lines during the other half frame period (field 2). For example, broadcast television and videotelephone systems in the United States transmit 30 frames per second using 2:1 interlace (60 fields per second). A 50 percent reduction in bit-rate can then be obtained by transmitting only 15 frames per second and displaying each frame twice as shown in Figure 2; however, jerkiness is then noticeable if the scene contains objects moving with moderate or rapid speed.

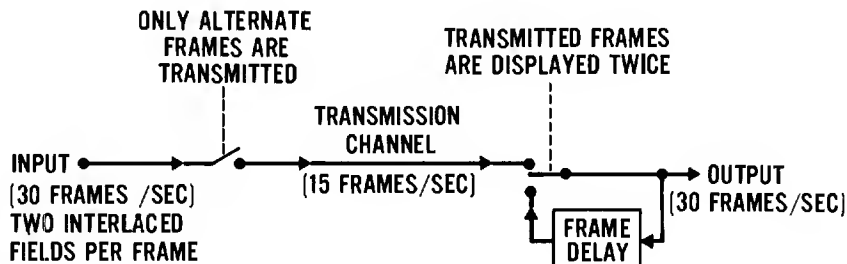


Figure 2 2:1 frame repeating. Reducing the frame transmission rate by half (15 Frames/sec) and displaying each frame twice at the receiver (30 Frames/sec) works well for movement at slow speeds but gives rise to visible jerkiness during movement at higher speeds.

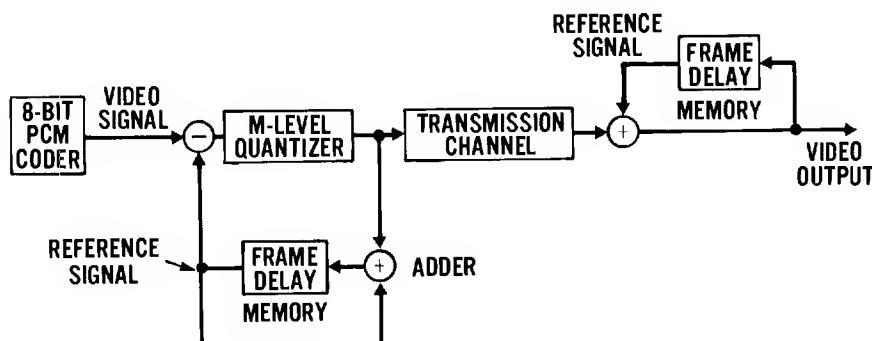


Figure 3 Frame-differential PCM. When transmitting frame-to-frame differences small values are quantized finely to give high quality reproduction in stationary areas of the picture, and large values are quantized coarsely to take advantage of the viewer's tolerance of brightness errors near moving edges in the picture. Because of the feedback configuration, a high quality picture is displayed within a few frame periods after motion in the scene has ceased.

If in Figure 1 the delays of the feedback coder and decoder are increased to one frame period, then the system transmits companded frame-differences as shown in Figure 3. Since small frame differences are finely quantized, scenes containing little or no movement are displayed with exceptionally high picture quality. When movement does occur, thereby creating larger, more coarsely quantized frame differences, some distortion is produced; but it is all confined to the moving areas of the picture where it is less noticeable to the viewer. Because of the feedback configuration, a high quality picture is again displayed within a few frame periods after motion in the scene ceases.

In this system a digital transmission error caused by the channel introduces an erroneous picture element into the receiver frame memory where it would remain indefinitely if some means for removing it were not provided. One method for doing this (called leak) is to reduce the magnitude of the reference signals by a small percentage both at the transmitter and at the receiver. The effect of a transmission error then slowly dies away as time progresses. Another technique which is more useful in interframe coders (called forced updating) is to transmit a small percentage of the picture elements in each frame as PCM. In this case a transmission error will remain visible until the corresponding PCM updating information arrives, at which time the error will suddenly disappear.

For video signals, between 30 and 60 levels of frame difference quantization are required for good picture quality during rapid motion.⁽⁸⁾ However, if picture quality during periods of movement is of no concern, then two-level quantization, i.e., one bit per element or frame-to-frame delta-modulation, is sufficient to reproduce still scenes such as charts or graphs with excellent fidelity.⁽⁹⁾ Since more quantization levels are required for frame differences than for element differences (for approximately equivalent picture quality during periods of rapid movement) there is little advantage in frame-differential coding unless multilength binary code words are used along with fairly large buffers. And if complex operations such as these are to be carried out, other techniques can be used which are of the same order of complexity but are much more efficient in terms of bit-rate reduction.



Figure 4 a) Videotelephone scene containing movement.
 b) Significant changes in a videotelephone scene containing moderate motion. The white picture elements denote frame-to-frame differences which are larger in magnitude than 1.5 percent of maximum. There are about 10,000 changes in this frame out of a total of about 56,000 visible elements.
 c) The white picture elements here are the ones which would be transmitted using cluster coding. For each cluster, an address is transmitted corresponding to the first picture element; then the frame differences are sent followed by a special code signalling the end of the cluster. Isolated changes in Figure 4b are ignored, and gaps of six picture elements or less between clusters are bridged to avoid sending excessive addressing bits.

Conditional Replenishment of the Reference Signal

In most videotelephone scenes, for example, the amount of movement is fairly small (see Figure 4b). There is very little camera panning, and there are few sudden scene changes. Furthermore, when there is a lot of movement such as when the camera is moved or when graphical material is being positioned, it is not necessary to transmit the picture with as high quality as in the normal situation. Thus, since there are usually so many small frame differences, there is considerable advantage in not transmitting them and, instead, sending only the frame differences which are significant; (10,11) that is, leaving the reference signals in Figure 3 unchanged unless the frame difference is large enough to warrant replenishment with new information. If only the significant frame differences are to be transmitted, i.e., those larger than about 1.5 percent of maximum, then addressing information must also be sent to tell the decoder where in the picture the received frame differences should be placed. Also, since data is generated at a very irregular rate depending on the location and on the amount of moving area in the picture, an elastic digital store or buffer must be used at the transmitter and receiver in order to use a constant bit-rate digital channel.

Conditional replenishment takes advantage of the long runs of insignificant differences that normally occur in videotelephone pictures. It has been found that significant frame differences also tend to occur in clusters, (8) and for this reason it is more efficient to address only the beginning and end of each cluster than to address each significant change separately. Moreover, since an address generally requires many more bits for transmission than does a frame difference, it is worthwhile to coalesce clusters of significant changes which are separated only by small gaps as well as to eliminate as much as possible isolated differences due to noise. The white dots in Figure 4b show the significant changes; Figure 4c shows the result of removing isolated changes and of bridging small gaps between clusters of changes.

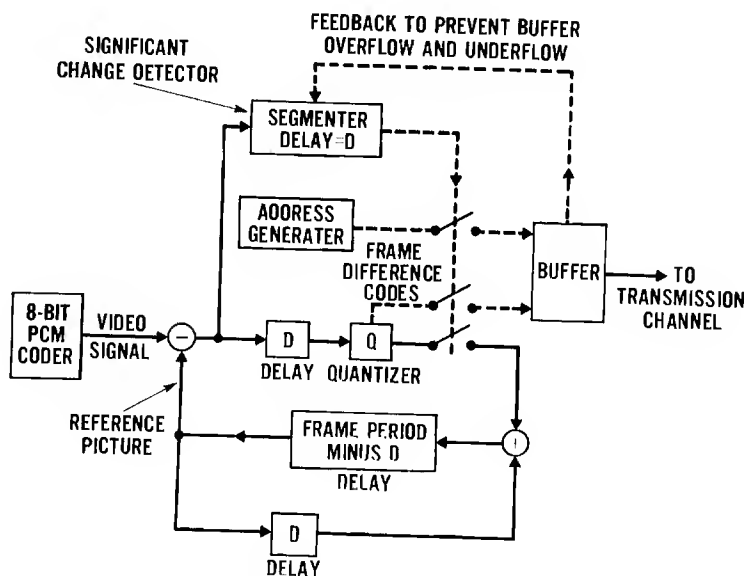


Figure 5 A conditional replenishment coder which transmits clusters of significant frame differences. The segmenter segments each frame into changed areas and stationary background areas, i.e., defines the clusters of frame differences which must be transmitted. The reference picture, which is available also at the receiver, is updated or replenished with quantized significant frame differences as determined by the segmenter. Since the segmenter requires a delay in order to operate efficiently, compensating delays must be included in the coder loops. Buffer overflow or underflow can be prevented by reducing or increasing the number of clusters transmitted. Forced updating (transmitting a few lines of each frame as PCM) to alleviate the effect of channel errors is not shown.

In the conditional replenishment coder of Figure 5 the segmenter defines the clusters of frame differences which must be transmitted. The required channel bit-rate depends on the size of the buffer and on the amount of motion in the picture which is to be accommodated. During periods of rapid movement when the data generation rate of the coder exceeds the channel rate, the buffer will begin to fill, and if this condition persists for too long, the buffer will overflow and data will be lost. Figure 6 shows

the required buffer size versus transmission bit-rate for a videotelephone picture in which a person is engaged in normal conversation. Sixteen bits are assumed for addressing each cluster and four bits on the average are assumed for each frame-difference. For high quality pictures (30-60 level frame difference quantization) multiword-length coding of frame differences prior to transmission is implied. The videotelephone signal format used here is: 1 MHz analog bandwidth, 271 lines/frame, 30 frames/sec with two interlaced fields per frame. Thus, 3 Megabits/sec corresponds to an average of 1.5 bits per picture element, blanking included.

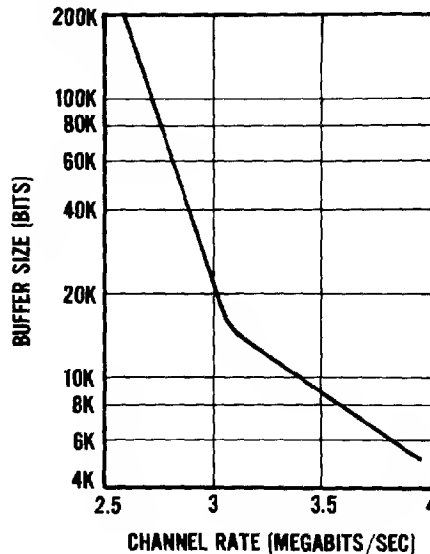


Figure 6 Buffer size versus transmission bit rate required for simple conditional replenishment where the scene contains the head and shoulders view of a person engaged in normal conversation. The curve is characterized by a sharp knee at the point where the buffer is just large enough to smooth the generated data over a field period (1/60 sec). Sixteen bits per cluster was assumed for addressing and the four bits on the average was assumed per frame difference. Very large buffers (several million bits) are capable of smoothing between one human action and the next (the fidget interval), but they introduce too much delay into the conversation to be useful.

Figure 6 shows that significant savings in bit-rate are achieved as the buffer size is increased up to about 15,000 bits. Around this point, data peaks generated in small areas of the picture are smoothed over an entire field period. Above 15,000 bits, however, the buffer begins to smooth the data over periods of time longer than a field interval (1/60 second here), and since the amounts of data in successive fields tend to be very much the same, relatively little saving of transmission bit-rate is obtained.⁽¹¹⁾ Very large buffers (several million bits) are capable of reducing the bit-rate significantly by smoothing the data from one human action to the next; however, besides being uneconomical they also introduce several seconds of delay into the transmission, and for face-to-face conversations over videotelephone, for example, this is intolerable. More will be said about smoothing the irregular data-rate when the subject of channel sharing by several coders is discussed.

Thus, for simple conditional replenishment there seems to be little advantage in providing more buffer capacity than is required to smooth the data over a field period. If normal motion is to be accommodated in head and shoulders views of people in conversation then with a 15,000 bit buffer, a transmission rate of about 3 Megabits/sec or 1.5 bits per picture element (including blanking) is needed. This is a saving of about 5:1 over 8-bit PCM. With a 600,000 bit buffer (which is about as large as can be used without introducing annoying delay into the conversation) a transmission rate of about 1.25 bits per picture element can be achieved. However, lower transmission rates can be achieved without using large buffers by using adaptive or multimode coding. These techniques are described next.

Adaptive or Multimode Conditional Replenishment

Viewers are accustomed to blurring in areas of the picture containing moderate or rapid movement. The main reason for this blurring is that since television cameras have no shutters, the light falling on a particular point of the target is integrated over one

frame period before being read out, and when there is movement this results in some blurring in the picture even before transmission.

Since rapidly moving objects are blurred, there is no need to transmit these areas with full picture resolution.^(13, 14, 8) Thus, a different mode of conditional replenishment coding can be used during periods of rapid movement. Since the transmission channel capacity required for conditional replenishment coders is determined mainly by the bit-rate generated during rapid motion, reducing the bit-rate during these periods lowers the required bit-rate of the channel.

A particularly simple method of reducing the moving-area resolution and, hence, the bit-rate in conditional replenishment systems is to transmit only every other significant frame difference along a line instead of all of them as is normally done. The moving-area picture elements which are not transmitted can then be obtained from the ones that are by simple interpolation. Full resolution is maintained in the stationary background areas of the picture since they are simply repeated from the previous frame. An effective way of making sure that the reduced resolution coding mode is only used during periods of rapid movement is to bring it in only when the number of bits stored in the buffer exceeds a certain threshold value. Subsampling the moving area at half-rate in this way almost halves the channel capacity required for transmission. It has been found that further horizontal subsampling gives rise to visible degradation except with extremely rapid motion.⁽¹⁵⁾

Feedback from the buffer can be used to control the picture quality in other ways as well. For example, the frame difference threshold value used by the segmenter to test for significance can be raised as the buffer fills. And when there is violent motion or movement of the camera (periods during which the viewer is not very critical of picture quality) a simple, but effective way of cutting down the data rate is to stop conditional replenishment altogether for one frame period whenever the buffer threatens to overflow and in this way give the buffer a chance to empty. At the receiver the previous frame is simply displayed twice resulting in a slight jerkiness in the movement instead of the smooth motion which is normally reproduced.

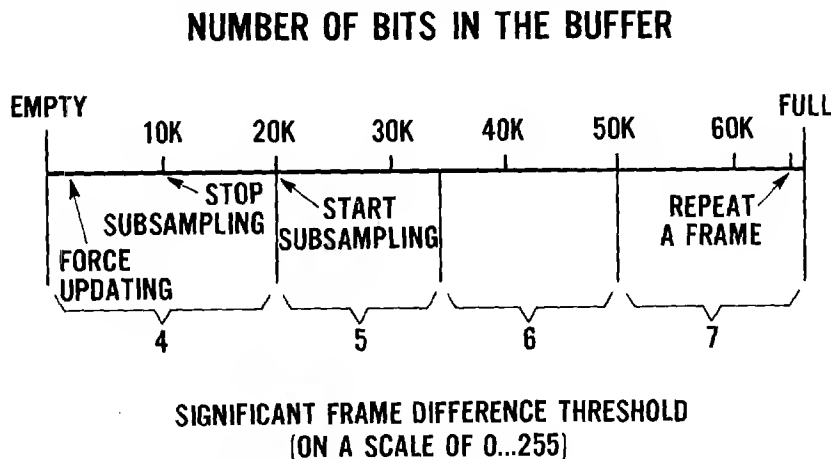


Figure 7 Multimode conditional replenishment where the coding mode is controlled by the buffer fullness or queue length. Fairly active motion can be accommodated with a 65K bit buffer and a transmission channel rate of 2 Megabits/sec or 1 bit per picture element. A saving of 8:1 over 8-bit PCM is obtained. The significant frame difference threshold is raised from 4 to 5, 6 and 7 on a scale of 0...255 as the buffer-queue-length increases beyond 20K, 35K and 50K bits, respectively. All replenishment is stopped when the queue length exceeds 65K bits, and it does not resume again until the queue length falls below 2500 bits. The system starts subsampling in the moving area at half rate when the queue length exceeds 20K bits and does not return to full sampling until the queue length falls below 10K bits. One TV line is sent as 8-bit PCM (forced updating) whenever the queue length falls below 2500 bits.

Multimode conditional replenishment with forced updating, subsampling at half rate during rapid movement and frame repeating to prevent buffer overflow as shown in Figure 7 requires, for videotelephone, a transmission rate of about 2 Megabits/sec - an 8:1 saving in bit rate over 8-bit PCM.⁽⁶⁾ Fairly active motion can be accommodated using a buffer size of about 65K bits. Round trip delay introduced by the coder is about 1/15

of a second which is not noticeable in most situations. If, during violent motion, the transmitting buffer fills then repeating one frame allows the buffer to completely empty so that several frames of violent motion can then be transmitted before another frame must be repeated. Emptying of the transmitter buffer during periods of very slow motion can be conveniently prevented by a forced updating of one TV line as 8-bit PCM whenever the number of bits stored in the buffer falls below a certain threshold. In fact, it is prudent to transmit a few lines of each frame as PCM regardless of the state of the buffer in order to accommodate for the effects of digital transmission errors which would otherwise last indefinitely.

Conditional Subsampling in the Vertical Direction

Subsampling horizontally along a line in the moving area at less than half rate gives rise to visible degradation unless movement is very fast.⁽¹⁴⁾ However, moving-area subsampling at half rate in the vertical direction is feasible, and like horizontal subsampling it reduces the transmitted data-rate significantly. Because of interlace, vertical subsampling at half rate in the moving area is the same as not transmitting any frame differences at all during alternate fields. The moving area picture elements in those fields would be obtained at the receiver by interpolation between the adjacent fields⁽¹⁵⁾ as shown in Figure 8. The effect on spatial resolution is usually negligible partly because of the smearing due to movement and partly due to the fact that good vertical resolution is difficult to obtain in a television picture in any case because of the scanning mechanism.

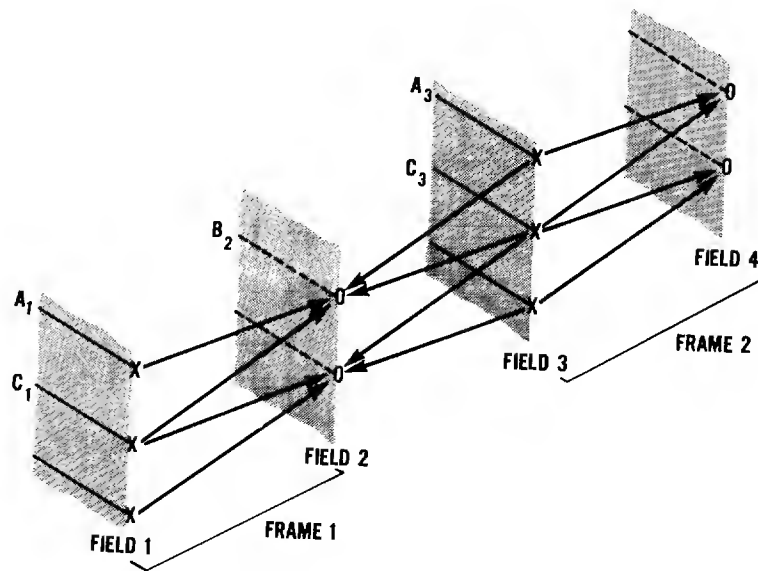


Figure 8 Subsampling at half rate in the vertical direction. Because of interlace this is equivalent to conditionally replenishing alternate fields only. In the diagram, fields 1 and 3 would be coded and transmitted via conditional replenishment. Moving area picture elements in the intervening fields would be obtained at the receiver by interpolation. Thus, changed picture elements in line B₂ would be obtained by a four-way average of picture elements in lines A₁, C₁, A₃ and C₃. Simple vertical subsampling and four-way interpolation in this manner does not give good rendition of moving edges, however, and results in a slightly visible jerkiness in the picture. By sending a small amount of additional information to repair these edges, the jerkiness can be eliminated. This is called conditional vertical subsampling.⁽¹⁶⁾

The effect of field-to-field interpolation does have a noticeable effect on temporal resolution, however, especially during periods of rapid motion.^(15,16) Since the interpolation is based on picture information occurring 1/60 of a second away in time, moving edges are poorly represented, and this leads to jerkiness in movement rendition when large areas move fast. This jerkiness is much less noticeable than with frame repeating, but it is nevertheless detectable.

In most scenes only a small amount of additional information need be transmitted to the receiver to enable it to adequately reproduce those edges which have been distorted by the field-to-field interpolation.⁽¹⁶⁾ This procedure is called "conditional vertical subsampling." The white picture elements of Figure 9 show where simple field-to-field

interpolation is not good enough and additional information must be transmitted. The original scene is the same as in Figure 4. In a single videotelephone field of Figure 4c there are 7870 picture elements in the moving area, but only 2250 of them as shown in Figure 9 need be transmitted using conditional vertical subsampling. The channel bit-rate can be reduced by about a factor of 1.5 using conditional vertical subsampling.

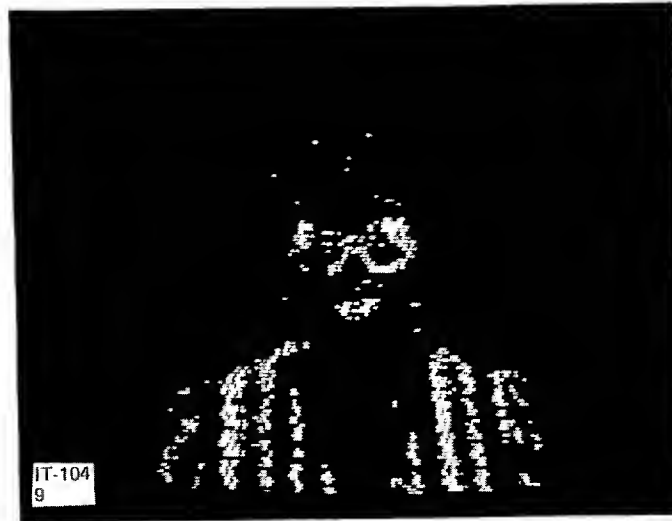


Figure 9 Picture elements in one field of Figure 4 which would be transmitted using conditional vertical subsampling with threshold $T = 12$ (on a scale of 0...255). In the fields to be interpolated (fields 2 and 4 of Figure 8) the actual value of each moving area picture element is compared with the value obtained by the four-way interpolation. If the magnitude of the difference exceeds the threshold T , then the difference exceeds the threshold T , then the difference is transmitted along with addressing information. Otherwise, nothing is sent, and the four-way average is used at the receiver. In this field, there are 7870 picture elements in the moving area, but only 2250 need be transmitted using conditional vertical subsampling.

Buffer and Channel Sharing

The channel bit-rate required by the conditional replenishment coder of Figure 5 is determined mainly by the peak data-rate produced by the coder, i.e., the data-rate generated during periods of active motion in the scene. Since periods of active motion in videotelephone scenes, for example, tend to come in bursts, separated by long intervals of inactivity, the channel bit-rate tends to be much larger than the long-term average data-rate (excluding the forced-updating needed to prevent buffer emptying). As was stated previously, an extremely large buffer (several million bits) would be needed in order to appreciably smooth the data from one period of active motion to the next and, thus, bring the channel bit-rate down closer to the long-term average data-rate. Such a buffer would not be feasible because of economic reasons and because of the excessive delay it would introduce into a videotelephone conversation.

At any given time, however, the data-rate averaged amongst many conditional replenishment coders will be fairly close to the long-term average data-rate produced by a single coder. Thus, if the data produced by many videotelephone users is combined prior to buffering and transmission, as shown in Figure 10, the advantage of a large buffer can be realized, namely lower channel bit-rate per data source, without the disadvantages of high buffer-memory cost and excessive delay. This technique automatically allocates more buffer space and channel capacity to those users which are temporarily in rapid motion at the expense of those which are not. Computer simulations have been carried out to evaluate this scheme, and results indicate that by combining the data from about 15 conditional replenishment coders prior to transmission, the channel bit-rate per coder can be halved and the buffer size requirements can be drastically reduced. (19)

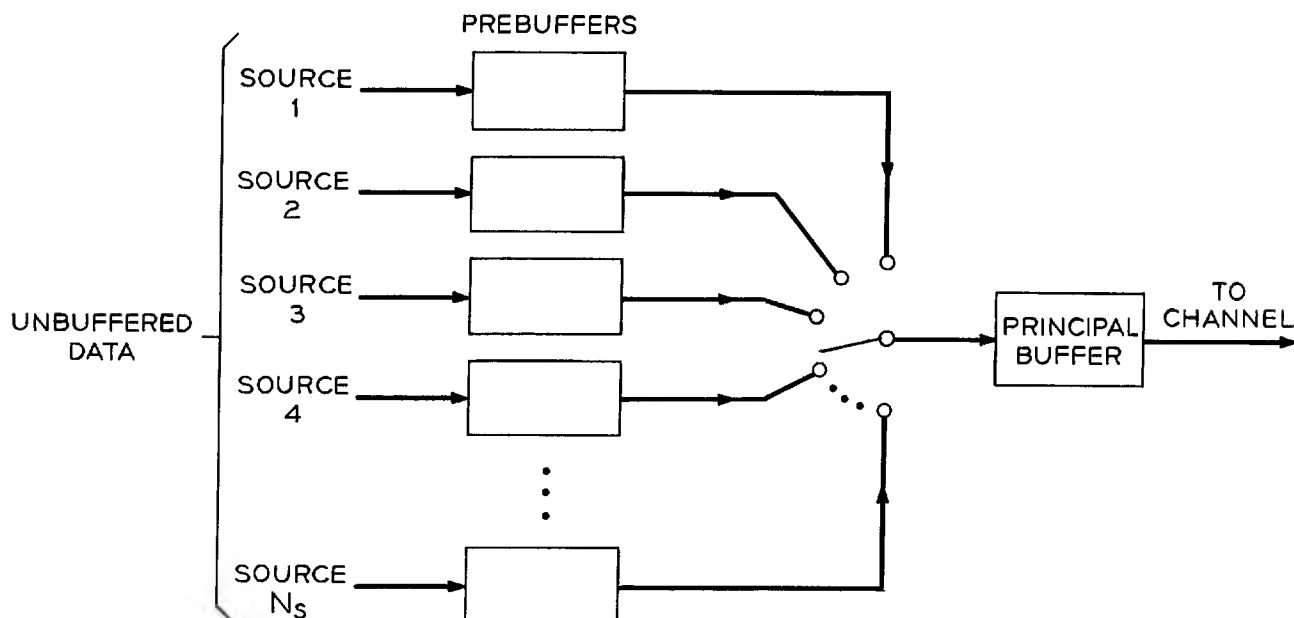


Figure 10 Buffer and channel sharing by several conditional replenishment coders. The unbuffered data, (entering from the left) produced by each coder occurs at a very irregular rate - short periods of high data-rate caused by active motion in the videotelephone scene tend to be separated by long periods of low data-rate. The data is first stored temporarily in individual prebuffers, then transferred to the principal buffer via the multiplexing switch, and finally transmitted over a single high capacity channel to the receiver where the inverse operation takes place. This technique automatically allocates more buffer space and channel capacity to those sources which are temporarily producing the higher data-rates. By sharing the channel in this way amongst about 15 conditional replenishment coders, the channel bit-rate per coder can be halved.⁽¹⁹⁾

Frame-to-Frame Coding of Signals Which Have Been Previously Transmitted Via Intraframe Coding

A possible configuration for long-distance transmission of video signals is the hierarchical arrangement shown in Figure 11. Within large cities, for example, the signals might be sent via relatively cheap intraframe coding techniques such as element differential PCM. Between cities where transmission costs are much higher, interframe techniques such as those described previously might be used to reduce the bit-rate as much as possible.

A fundamental difficulty arises with this situation, however. All intraframe coders introduce a certain amount of frame-to-frame noise values into the signal which, although they may not be objectionable to the viewer, are detected by the interframe coder as a large number of significant frame differences. The white picture elements in Figure 12b show the frame differences which are larger in magnitude than 1.5 percent of maximum using a video signal which has been passed through an element-differential PCM codec. The number of significant changes far exceeds the number of changes which occur when 8-bit PCM is used as the input signal as in Figure 4b.

In order to reduce the number of picture elements which must be transmitted it is not sufficient to simply raise the threshold which determines whether or not a frame difference is significant. Unacceptable picture quality results. Instead, advantage must be taken of the distinguishing properties of the frame differences caused by intraframe quantization noise as opposed to the distinguishing properties of the frame differences caused by movement. For example, frame differences caused by intraframe quantization noise tend to be uncorrelated both spatially and temporally, and they are usually larger near edges in the picture because of companding. Frame differences caused by movement, on the other hand, are highly correlated both spatially and temporally and tend to be large only near moving edges in the picture.

Using information such as this, algorithms have been designed for segmenting the picture with good accuracy into moving areas and stationary background areas.⁽¹⁷⁾ The white

picture elements in Figure 12c show the moving areas as defined by such an algorithm for the same signal used in Figure 12b. However, the number of moving-area picture elements still tends to be significantly higher than with an 8-bit PCM input. By using conditional vertical subsampling at half rate, horizontal subsampling at half rate and multiword-length coding of frame differences, a multimode conditional replenishment coder for videotelephone has been simulated which operates with an average data transmission rate of 2 Megabits/sec (1 bit/picture element). (18)

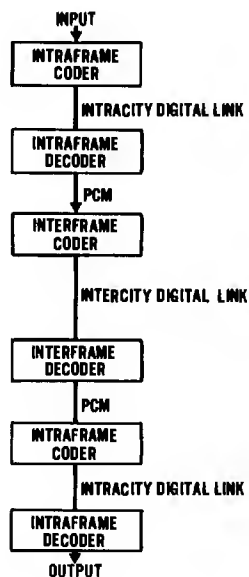
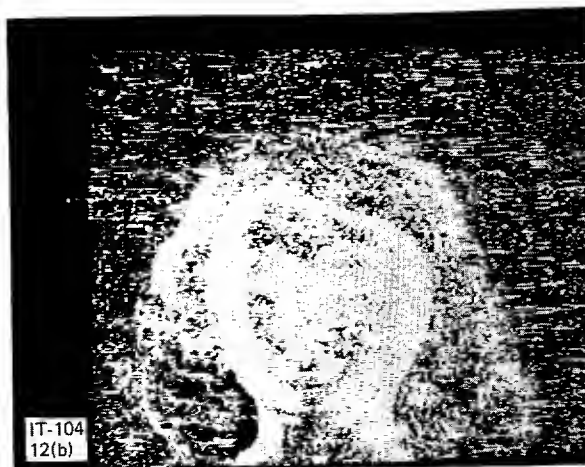


Figure 11 A possible configuration for long-distance transmission of videotelephone signals. Cheaper intraframe codecs are used for short, intracity digital links. Between cities where transmission costs are high, interframe codecs are used to reduce bit-rate as much as possible.

Coding Method	Channel Bit-Rate Megabits/Sec
1. 8-Bit PCM	16
2. Element-Differential PCM	8
3. Simple Conditional Replenishment	3
4. Multimode Conditional Replenishment	2
5. Multimode Conditional Replenishment and Conditional Vertical Subsampling	1.3
6. Multimode Conditional Replenishment, Conditional Vertical Subsampling and Channel Sharing by 15 Users	0.7

TABLE I - Bit-rates required for videotelephone using various coding techniques. Element-differential PCM is an intraframe coding technique while methods 3-6 are interframe techniques. Method 4 has been demonstrated while the figures cited for methods 5 and 6 are based on simulations.



- Figure 12 a) Videotelephone scene containing movement.
- b) The white picture elements denote frame-to-frame differences which are larger in magnitude than 1.5 percent of maximum using a video signal which has been passed through an element-differential PCM codec. The number of significant changes far exceeds the number of changes in Figure 4c where 8-bit PCM was used as the input signal.
- c) Using the same video signal as in a) the white picture elements here denote the moving area as defined by a more sophisticated segmenting algorithm.⁽¹⁷⁾ This algorithm takes advantage of a number of signal properties which enable it to distinguish between frame differences due to intraframe quantization noise and the frame differences caused by movement.

Conclusion

This article has covered some of the known methods for taking advantage of frame-to-frame redundancy in video signals in order to reduce the transmission channel bit-rate. The problem was approached from two sides. First, since only a portion of the picture elements change significantly between frames, there is considerable advantage in transmitting only the changed area. i.e., using conditional replenishment.^(10,11) Second, since viewers are less critical of picture quality in moving areas than in the stationary areas, a saving of bit-rate can be achieved by progressively lowering the resolution of the moving area as the amount of movement increases.^(13,8,14)

The bit-rate savings possible using various frame-to-frame coding techniques on videotelephone signals are shown approximately in Table I. A saving of 8:1 in bit-rate over 8-bit PCM has been demonstrated using multimode conditional replenishment,⁽⁸⁾ while

the figures cited for conditional vertical subsampling⁽¹⁶⁾ and channel sharing⁽¹⁹⁾ are based on simulations. All but rapid movement of large areas is accommodated with excellent picture quality. With large area movement, frame repeating is used which causes some visible jerkiness in the picture. If the input signal contains noise due to a previous intraframe coding and decoding, then too many spurious frame differences are present to use only these methods. More sophisticated moving area detection techniques⁽¹⁷⁾ must be used to transmit such a signal efficiently.

Transmission errors in the digital channel have been mentioned only in passing. A basic principle in the efficient coding of information is that the more one removes redundancy from a data stream the more vulnerable to transmission errors the information becomes. This is certainly true of picture coding. However, some information is more important to the receiver than other information. For example, in conditional replenishment, addresses and synchronization information are far more important than frame differences. Thus they should be protected by adding redundancy in a controlled fashion. In this way, the effect of transmission errors can be minimized.

Other techniques for removing frame-to-frame redundancy are currently being explored.⁽²⁰⁻²²⁾ However, most of these schemes, like the techniques described in this paper, are fairly expensive to implement, and this tends to limit their usefulness at the present time to only long-distance transmission links where the expense of the redundancy reduction systems are significantly outweighed by the savings in transmission costs. In the future, however, prices of digital logic and storage are expected to continue the dramatic decline experienced in the last several years making some of these bit-rate reduction schemes economically feasible for digital transmission over shorter distances.

Acknowledgment

This article describes the work of a number of people, many of whom we have had the opportunity to talk with personally. We would like to express our gratitude to them for the many stimulating discussions which we have had.

REFERENCES

1. Special Issue on Redundancy Reduction, Proceedings of the IEEE, v. 55, no. 3, March 1967.
2. Special Issue on Digital Picture Processing, Proceedings of the IEEE, v. 60, pp. 768-820, July 1972.
3. G. Lapidus, "The Domain of Magnetic Bubbles," IEEE Spectrum, v. 9, no. 9, September 1972, pp. 58-62.
4. W. S. Boyle and G. E. Smith, "Charge-Coupled Devices - A New Approach to MIS Device Structures," IEEE Spectrum, v. 8, no. 7, July 1971, pp. 18-27.
5. L. M. Terman, "MOSFET Memory Circuits," Proceedings of the IEEE, v. 59, no. 7, July 1971, pp. 1044-1058.
6. G. S. Kino and H. Matthews, "Signal Processing in Acoustic Surface-Wave Devices," IEEE Spectrum, v. 8, no. 8, August 1971, pp. 22-35.
7. R. C. Brainard, F. W. Mounts, and B. Prasada; "Low Resolution TV: Subjective Effects of Frame Repetition and Picture Replenishment," Bell System Tech. J., v. 46, pp. 261-271, January 1967.
8. J. C. Candy, M. A. Franke, B. G. Haskell, and F. W. Mounts, "Transmitting Television as Clusters of Frame-to-Frame Differences," Bell System Tech. J., v. 50, pp. 1877-1888, July-August 1971.
9. R. Schaphorst, "Frame-to-Frame Coding of NTSC Color TV," in Symposium on Picture Bandwidth Compression (Mass. Inst. Technol., Cambridge, Mass., April 1969).
10. A. J. Syler, "Real-Time Recording of Television Frame Difference Areas," Proc. IEEE (Corresp.) v. 51, pp. 478-480, March 1963.
11. F. W. Mounts, "Video Encoding System with Conditional Picture Element Replenishment," Bell System Tech. J., v. 48, pp. 2545-2554, September 1969.
12. J. O. Limb, "Buffering of Data Generated by the Coding of Moving Images," Bell System Tech. J., v. 51, pp. 239-259, January 1972.

13. A. J. Seyler, "The Coding of Visual Signals to Reduce Channel-Capacity Requirements," Proc. IEEE, v. 109, pt. C, pp. 676-684, September 1962.
14. R. F. W. Pease and J. O. Limb, "Exchange of Spatial and Temporal Resolution in Television Coding," Bell System Tech. J., v. 50, pp. 191-200, January 1971.
15. J. O. Limb and R. F. W. Pease, "A Simple Interframe Coder for Video Telephony," Bell System Tech. J., v. 50, pp. 1877-1888, July-August 1971.
16. R. F. W. Pease, "Conditional Vertical Subsampling - A Technique to Assist in the Coding of Television Signals," Bell System Tech. J., v. 51, pp. 787-802, April 1972.
17. D. J. Connor, J. O. Limb, R. F. W. Pease, and W. G. Scholes, "Detecting the Moving Area in Noisy Video Signals," to be published.
18. D. J. Connor, B. G. Haskell, and F. W. Mounts, "A Frame-to-Frame PICTUREPHONE Coder for Signals Containing Differential Quantizing Noise," Bell System Tech. J., v. 52, pp. 35-51, January 1973.
19. B. G. Haskell, "Buffer and Channel Sharing by Several Interframe PICTUREPHONE Coders," Bell System Tech. J., v. 51, pp. 261-289, January 1972.
20. B. G. Haskell and J. O. Limb, "Predictive Video Encoding Using Measured Subject Velocity," U. S. Patent 3,632,865; January 4, 1972.
21. B. G. Haskell, "Frame-to-Frame Predictive Coding of Videotelephone Signals," IEEE International Conference on Communications - Conference Record, v. 8, p. 31-51, June 19-21, 1972.
22. F. Rocca and S. Zanoletti, "Bandwidth Reduction Via Movement Compensation on a Model of the Random Video Process," IEEE Trans. on Communications, v. COM-20, pp. 960-965, October 1972.

BANDWIDTH COMPRESSION OF MULTISPECTRAL DATA

A. Habibi and A. S. Samulon
TRW Systems Group
Redondo Beach, California 90278

Abstract

The coding methods suitable for bandwidth compression of multispectral imagery are considered. These methods are compared using various criteria of optimality such as MSE, signal-to-noise ratio, recognition accuracy, as well as computational complexity. Based on these results the candidate methods are reduced to three recommended methods. These are KL-2 Dimensional DPCM, KL-cosine-DPCM and a cluster coding method. The performance of the recommended methods are examined in the presence of a noisy channel and concatenated with a variable rate (Huffman) encoder.

Introduction

Improvements in earth resource satellite MSS scanners will result in higher spatial resolutions, greater number of spectral bands, and faster output rates for the multispectral earth resources data. This will require larger bandwidth for transmission of the multispectral data as well as larger storage capacities and more complicated ground processing systems for classification and dissemination of the data. Compression of data at the source can reduce the costs and the constraints upon each of these parts of a scanner-to-user information management system. Most studies of data compression techniques for multispectral data have considered only one coding technique. General Electric and Philco-Ford have considered only simple and elementary coding algorithms [1-2]. Purdue University has considered use of transform coding techniques and clustering algorithms for compressing the multispectral data in separate studies [3,4]. TRW Systems Group considered coding multispectral data using errorless entropy coding of this data for archiving applications [5]. Jet Propulsion Laboratory has considered using clustering algorithms for joint bandwidth compression and classification of multispectral images [6]. These studies have used different data bases. They leave out many types of efficient coding algorithms, and generally do not include the effect of important parameters such as channel noise and the sensor behavior on the performance of the individual coding algorithms. In addition due to use of different data bases and different criteria of optimality a meaningful comparison of various coding techniques is impossible.

We consider a number of promising algorithms for on-board compression of the multispectral data. Included are Differential Pulse Code Modulator (DPCM) and transform coding systems as well as hybrid coding techniques. The performance of these systems are improved by concatenating these coding algorithms with variable rate (Huffman) encoders. In addition, we examine a cluster coding method which reduces the bandwidth of multispectral imagery by classifying the multispectral data into a fixed number of clusters. The clusters are approximated at the receiver using a classified image and the cluster averages. These algorithms are compared based on their performances on representative ERTS data. The experimental results are obtained by simulating the algorithms on digital computers. A number of criteria of optimality such as mean square error, peak-to-peak signal to noise ratio, retention of classification accuracy and the computational complexity are used to compare these techniques.

Criteria of Bandwidth Comparison

Several criteria serve as a basis for comparison of alternative compression techniques. These include distortion, complexity, and susceptibility to channel and sensor effects.

Distortion

In order that data compression be possible, one or both of two conditions must be satisfied. The first condition is that the data have some structure or redundancy. By knowing something about the data structure, data compression can be accomplished without losing any of the information the data contains. Hence, data compression that relies solely on knowledge of data structure is called information preserving or error free data compression. The second condition which leads to compression is that the image user have interest in only limited accuracy or a particular aspect of the data collected. In this case, data which is of low value to the user can be discarded, resulting in data compression. Since information discarded is permanently lost, compression based on user interest is called information destroying data compression.

In order to determine whether a particular information destroying compression algorithm preserves sufficient information for a particular use, a criterion must be defined by which the information loss can be measured. Because of its mathematical convenience, a number of authors have used mean square error as a measure of distortion. We use this as one measure during our study. It has the advantages of being widely used and understood, of making tractable analytical treatments of the problem, and of leading to satisfactory results for many applications. On the negative side, it is a criterion not specific to particular uses of the data.

A criterion closely related to mean square error is the signal to noise ratio. Indeed this criterion can be considered a normalized form of the MSE. Peak-to-peak signal to root mean square (RMS) value of the noise as well as RMS signal-to-noise ratio is widely used by the television industry as a measure of television signal quality.

Recognition Accuracy

Another criterion which is more application oriented is the algorithm's effect on classification accuracy. For example, if each raster point in the image is to be classified as corresponding to some crop or land use, based on the recorded brightnesses in the several spectral bands, the increase in classification errors resulting from performing classification on the reconstruction of the compressed data rather than on the original image is a measure of the relevant information lost due to compression.

The increased probability of misclassification can be used as a distortion measure, but, unfortunately, the complexity of the classification procedure generally precludes calculation of the rate distortion function.

The problem of classification is equivalent to partitioning the space of possible measurement values. All measurement values falling within one element of the partition are viewed as representative of some category. For example, in the case of earth resources data, one might assume that all recorded brightness values within some range are representative of wheat.

Two basic approaches exist for determining a partition. In one method, a set of categories is specified. The space of measurement values is then broken into segments such that the values in a particular segment correspond with one of the prespecified categories. The segments are chosen in a way that in some sense is "good". In the other approach to determination of a partition, no categories are specified in advance. Instead the measurement values are examined to see if they cluster into groups. In general it is difficult to define what is meant by certain values belong to one cluster and certain values belong to another cluster. A myriad of criteria have been proposed. Most of them yield different clusters, for some sets of measurement values, than a human would intuitively obtain.

Now we define what is meant by preservation of classification accuracy under a distortion of the measurement space. We assume that we have two measurement spaces, each having its own partition. In the first space are the undistorted measurements, x , and in the second are the distorted measurements. In order that classification accuracy be preserved the two partitions should be such that if y is the distorted value corresponding with x , then the category which y represents should be the same as the category which x represents. Using more precise notation, we make the following definitions:

- X a space of measurements with elements x
- P_X a partition of X ($P_X(x)$ is the category which x represents)
- Y a space of measurements with elements y
- P_Y a partition of Y
- D a mapping from X into Y

Then we say that P_Y preserves the classification accuracy of P_X under the measurement distortion D .

We can see in Figure 1 that class 1 and class 2 can be distinguished equally well in the space X and the space Y with the appropriate partitions. Under certain distortions, however, it may not be possible to preserve classification accuracy. It is always possible to find a partition which preserves classification accuracy. It is always possible to find a partition which preserves classification accuracy if D is a 1 to 1 function since, in that case, values which can be distinguished in space X can be distinguished in space Y . If D is not a 1 to 1 function, it may still be possible to preserve classification accuracy. For example a function which maps all values in one category in the original space into one point in the distorted space can still preserve classification accuracy as long as it maps values in different categories into different points.

We now fit the definition we have made into the context of earth resources data. For each point in the image we obtain a set of brightness measurements in several spectral bands. The space of measurement values in the various spectral bands can be represented by the space X in the definition above. In addition, further measurements can be obtained by taking into account the spatial variation of brightness in the neighborhood of each point. Such measurements are often called "texture" measurements. These measurements can also be included in the space X . Thus, for each point in the image we have a space X of measurement values. Using pattern classification, we can determine, based on the measurement space, what category (e.g., wheat, petroleum bearing land) is represented by a particular point in the measurement space. Data compression which is classification accuracy preserving is then obtained in two ways. First, several points in the same category can be mapped into one point. Second, spatial redundancy of the measurement values can be exploited to achieve further data compression.

It was decided to use a clustering technique for classification for several reasons. First, a technique using prespecified categories would have required training samples to define the best partition.

Thus, ground truth information would be required for each prespecified category represented in the data. Second, each category (e.g. winter wheat) has a different spectral signature depending on sun angle, latitude, season, and so on. Therefore, ground truth data for the particular data sets processed would be required. Third, we wanted to use a data set with a number of different categories, including a variety of crops, water, desert, and mountains so that any conclusions developed during the study would be relevant for a number of disciplines.

In the classification accuracy determination procedure, we calculate the centroids of the clusters. Unfortunately, calculation of the centroids is an iterative process which is quite time consuming. Hence, it is desirable to sample the available imagery and determine the centroids using a subset of the picture elements. For convenience we have chosen to use only elements belonging to every fourth row and column of the imagery in determining the centroids.

Using the clustered subset we obtain the centroids of the clusters. By centroid is meant the following. Each cluster has a particular mean value in every spectral band. The centroid of a cluster is the set of mean values for that cluster. Figure 2 shows pictorially the meaning of centroid for the case of only two spectral bands.

The centroid finding program has a number of steps that can be summarized as follows: A particular number of clusters is selected. More or less arbitrarily, a set of initial centroid values are chosen. Each sample value is assigned to the category corresponding to the closest centroid. Then the centroids of the sample values belonging to each class are computed. Now the data samples are mapped into those categories having the closest of the new centroids. This entire process is repeated until very few of the sample values change category on one of the iterations. At that point, the so called "Swain-Fu" distance between each pair of classes is measured. If the minimum distance between classes is less than some prescribed distance, the two closest classes are merged and the whole process is repeated with one fewer class. Otherwise the clustering algorithm terminates. A minimum number of clusters is also specified in order that merging not continue indefinitely.

After completing determination of the cluster centroids, the entire 256x256 image is classified. This is accomplished by determining to which centroid each sample is closest. The output of this program is an image in which the value corresponding with any pixel is the number of the cluster to which it belongs.

In order to compare the clusters obtained on the original imagery and on the reconstruction of the compressed imagery, we first determine the correspondence between clusters in the two sets of imagery. To illustrate what is meant by finding a correspondence between clusters, consider the case when cluster 1 in the original image represents water, and in the reconstruction water is represented by cluster 3. In order to save time this correspondence determining program does not find the cluster correspondence which will minimize the difference in classification but instead first minimizes the difference in classification for the subarea of pixels belonging to the cluster with the most elements, then minimizes the difference in classification error for the subarea of pixels belonging to the cluster with the next greatest number of elements, and so on.

Once the cluster correspondence is made the percentage of picture elements classified differently in the original imagery and the reconstruction is determined. This number is used as the classification inaccuracy for the particular bandwidth compression method.

Implementation Complexity

Once it is shown that several compression techniques can significantly reduce the required data rate without introducing unacceptable distortion, it is necessary to compare the implementation complexity of the techniques. Compressor complexity is a particularly significant factor for spacecraft applications. Reconstruction complexity on the ground has less effect on overall system costs.

To evaluate complexity, an algorithm was proposed for each technique. Then the required number of additions and multiplications were counted and the needed buffer sizes determined, based on typical block sizes and numbers of spectral bands.

Sensitivity to Sensor Effects

The properties of the sensors that gather multispectral earth resources data can be expected to affect the data structure and thus in turn affect compression performance. As a typical sensor to be used in the next decade for gathering multispectral earth resources data, we considered the thematic mapper projected for NASA's EOS. The thematic mapper is a multispectral scanner similar to that on the LANDSAT (ERTS). It uses photodiodes as transducers and a moving mirror to scan lines perpendicular to the satellites track. The sensor properties that can be expected to affect compression are nonuniformity and nonlinearity of the photodiodes, and geometric distortion and spectral misregistration due to the optics and scanning patterns.

For a fixed mean square error, the compression obtainable with a particular compression technique can be related to the correlation coefficient of the process of which the data is a sample function. Based on the scanning geometry of a likely thematic mapper and on the instruments present specifications, the variation in correlation coefficient over an entire image was predicted under the assumption that the data be samples of a wide sense stationary process. These variations were in turn related to increased mean square error or equivalently to decreased signal to noise ratio.

Bandwidth Compression Methods for Multispectral Imagery

Multispectral data can be modeled by a stochastic process $f(x,y,k)$ which is a function of three discrete variable x , y , and k ; x and y refer to spatial variables, and k is the spectral variable. In theoretical studies of image bandwidth compression methods one often models the data as a stochastic process having statistics identical to those of the imagery so that one can then evaluate the performance of various encoders on the process. This approach to the analysis of algorithms used for coding video data is often carried out by assuming that the stochastic model $f(x,y,k)$ is Markov on all three variables x , y , and k . This would imply that the auto-covariance of the assumed stationary process $f(x,y,k)$ is exponential in all three dimensions [3], i.e.

$$R(x,\hat{x},y,\hat{y},k,\hat{k}) = e^{-\alpha_1|x-\hat{x}|-\alpha_2|y-\hat{y}|-\alpha_3|k-\hat{k}|} \quad (1)$$

This is a desirable assumption since it simplifies the analysis of the problem to a considerable degree. The results are valid on most naturally obtained video data that show ordinary movements in both spatial and temporal directions. Considering this model for MSS data one proceeds with the correlation measurements which specify α_1, α_2 , and α_3 in (1). These measurements also verify the correctness and the accuracy of the above model.

Measurements of the spatial correlation of the data reveal that the correlation of the data in the horizontal and the vertical directions is indeed exponential. However, spectral correlation does not reduce exponentially by moving across the bands. Indeed various bands show similarities and differences which are totally different from interframe behavior of other video data encountered in television or in reconnaissance flights. The following summarizes the statistics of the MSS LANDSAT imagery.

- a) Large positive correlation between the red and the green bands.
- b) Large positive correlation between the two infrared bands.
- c) Small negative correlations between the red and the two infrared bands.

While these correlations apply on the average they do not apply for many particular classes - water being one counterexample.

Transform Coding

MSS data is three-dimensional, it possesses spectral as well as spatial correlation. Three-dimensional transform coding methods take advantage of the correlation of the data in all three-dimensions for bandwidth compression. In this study we will use three-dimensional Fourier, Cosine, Hadamard and Slant transforms.

Delta Modulators and Predictive Coding

In the class of predictive coding methods only DPCM systems are considered. Delta modulators are not considered because for bit rates higher than one bit per picture element they require sampling the analog signal at a higher sampling rate. Besides it has been shown that the performances of delta modulators and adaptive delta modulators are suboptimal to the performance of two-dimensional DPCM encoders for image data [8].

In addition to DPCM and transform coding techniques the hybrid encoders that combine transform coding systems with DPCM encoders have been shown to give good performance for both intra-and inter-frame coding of television signals [9-10]. These systems as applied to MSS data can be divided in three categories;

- 1) Systems that perform a two-dimensional transform of each band of the MSS data and use a bank of DPCM encoders to process the transformed data across the individual bands.
- 2) The systems that take a transform across the spectral bands and use a two-dimensional DPCM encoder to process the data in the spectrally transformed domain.
- 3) Systems that follow a transformation in the spectral domain of MSS data by a second transform in the horizontal direction (scan direction) and a DPCM encoder in the vertical direction.

The hybrid systems in category (1) are rejected because most multispectral data have a small number of bands and this does not allow DPCM systems to reach a steady state. In addition small correlation of the spectral bands and the absence of the Markovian characteristics limit the performance of the DPCM systems.

Cluster Coding of Multispectral Data

Many important uses of earth resources data rely heavily on the use of computerized classification and pattern recognition of the multispectral earth resources imagery. For many classification applications multispectral data is first used to obtain a clustered image. This presents an alternate approach to compressing the bandwidth of multispectral imagery consisting of classifying the multispectral data on-board the space vehicle and transmitting only the clustered imagery. The problem with this approach is that there are many users who are interested in information contained in the individual spectral bands as well as the classified imagery. For this reason in addition to the classified picture, which contains an arbitrary number of clusters, additional information should also be transmitted such that individual bands of the original multispectral data can be reconstructed with an arbitrarily small level of degradation using the classified imagery and the additional information.

Different approaches to this method of bandwidth compression has been discussed in recent literature [4,6]. In this study we will discuss the cluster coding algorithm that uses Swain-Fu distance in generating various clusters.

Coding Methodology

Classification of multispectral data discussed in a previous section is a method of quantization in the spectral domain. That is, a number of picture elements in the measurement domain are represented by a single point in the four-dimensional space. This is illustrated in Figure for a two-dimensional space. Thus, the combination of the clustered imagery and the centroids represent a quantized form of the multispectral data which can be used to reconstruct a quantized form of each individual band. Thus, reduction of a multispectral image to an image of clusters and centroids for each cluster actually corresponds to exploiting the spectral correlation of multispectral imagery. The spatial correlation inherent in the multispectral imagery is preserved, to some extent, in the form of spatial correlation in the classified imagery. Naturally in addition to spatial correlation of the classified imagery, correlation of the centroids can be used for further compression of the bandwidth of the multispectral data.

The cluster coding algorithm discussed here uses the following steps for encoding and decoding the multispectral data.

1. A multispectral image is divided into blocks of 16 by 16 picture elements in each band.
2. Using all spectral components each block is partitioned into a number of classes. A clustered image and centroids corresponding to each cluster are the output of this step in the coding operation. The number of clusters in each block can be fixed or it can be allowed to vary generating differing numbers of clusters for each block. The latter corresponds to an adaptive bandwidth compression technique that operates at a variable bit rate where the former could be made to operate at a fixed bit rate. Even when the number of clusters is allowed to vary from one block to the other the maximum and the minimum number of clusters in each block can be fixed. This sets a fixed upper-bound and a lower-bound on the output bit rate which can be used for a more effective control of a storage buffer that has to be used with any variable-rate encoder.
3. The receiver reconstructs each block of the multispectral image by generating the individual bands in each block from the clustered image and the corresponding centroids of those clusters. The procedure is to examine each point in the clustered image and specify to which class it belongs. Then individual bands corresponding to the particular picture location are reconstructed by choosing their values equal to the centroid of that particular class.

The centroids are real numbers. Thus, they must be quantized and encoded. We have used the rather inefficient (PCM) scheme for coding the centroids for the preliminary results reported here. However, the ij th block in clustered imagery can be coded using $[\log_2 C_{ij}]^+$ binary digits where C_{ij} is the number of clusters in block indexed by i, j and $[\log_2 C]^+$ is the smallest integer larger than $\log_2 C_{ij}$. Using $[\log_2 C]^+$ bits for coding the ij th block in the classified imagery is a rather inefficient use of binary digits. Since it does not exploit any part of the spatial correlation in the clustered imagery. Several methods for efficient coding of the clustered imagery exist. One is using contour tracing combined with statistical coding algorithms such as Huffman encoding of directionals and gray levels. The other is use of differential encoders combined with the statistical coding algorithms. These modifications reduce the bit rate required for transmitting the clustered image and the centroids and thus correspond to a further bandwidth compression of the multispectral imagery.

The average bit rate per sample per band R_{ij} for the ij th block of imagery for the cluster coding algorithm using PCM transmission of centroids C_{ij} and the clustered imagery is,

$$R_{ij} = \frac{[\log_2 C_{ij}]^+}{B} + \frac{P C_{ij}}{N^2} \quad (2)$$

where 2^P is the number of levels used in quantizing each centroid component, B is the number of bands and N^2 is the number of samples in each block.

Equation (2) shows that the bit rate is directly related to the number of clusters in each block and the number of levels that one uses for quantization of centroids. It is inversely related to the number of samples in each block and the number of spectral bands in the multispectral data. Since one has to allow for a large number of clusters in each block to recover the details of multispectral imagery the method becomes more efficient for a large number of spectral bands.

The idea of dividing the multispectral imagery into blocks of N^2 samples and cluster coding each block is not essential for the coding algorithm. Indeed if one is interested only in a classified image using the whole image as one block will result in a very efficient encoder using this procedure. However, to reconstruct details in various bands requires a large number of clusters, which increases the computational complexity of the system enormously. On the other hand using relatively small block sizes detailed information in various spectral bands can be reconstructed rather accurately using a moderate number of classes.

Experimental Results

Simulation of DPCM, transform, hybrid and clustering methods using the criteria discussed in previous sections leads to recommending the following three bandwidth compression techniques for multispectral imagery;

1. A hybrid method using Karhunen-Loeve (KL) transform in the spectral domain followed by a two-dimensional DPCM encoder.
2. A hybrid method using KL transform in the spectral domain followed by a cosine transform in the horizontal and a DPCM encoder in the vertical direction.
3. The cluster coding method.

In methods (1) and (2) Haar transform is the most suitable deterministic replacement for the KL transform. In (2) Hadamard or Slant transform can replace the cosine transform with almost unnoticeable effects on the performance of the technique. It also was found that eliminating the spectral transformation effects the performance of methods (1) and (2) by about 1.5dB, based on the results using the LANDSAT imagery shown on Figure 3.

In the following the recommended techniques are compared using various criteria of optimality such as MSE, signal-to-noise ratio, recognition accuracy, and computational complexity. The performance of methods (1) and (2) is also evaluated in presence of channel noise and a variable rate (Huffman) encoder which further reduces the output bit rate without effecting the system performance.

Comparison of the Recommended Techniques Based on MSE and Signal-to-Noise Ratio

Figure 4 shows the performance of the three-dimensional hybrid encoder using KL-Cosine transforms with the DPCM encoder in comparison with the other two recommended bandwidth compression systems using S/N as the criterion of optimality. One is the system that follows the Karhunen-Loeve Transform in the spectral domain with a two-dimensional DPCM encoder. The other is the cluster coding technique. It is shown that the performance of the encoder using KL transformation followed by a two-dimensional DPCM encoder performs significantly worse than the hybrid encoder (KL-Cosine-DPCM) at low bit rate. However, their performances are rather similar at higher bit rates (about 2 bits per sample per band). The performance of the cluster coding method is superior to that of the hybrid encoder at low bit rates. However, it gives inferior results at high bit rates. In this aspect its performance is completely opposite to that of the KL-2 Dimensional DPCM encoder.

Figure 4 also shows the performance of the hybrid (KL-Cosine-DPCM) and KL-2 Dimensional DPCM encoders in concatenation with a variable rate encoder. The variable rate encoder is the Huffman encoder which reduces the output bit rate by assigning shorter code words to more probable output levels and longer code words to the less probable output levels. The effect of entropy coding on the performance of the KL-Cosine-DPCM encoder is less than its effect on the performance of KL-2 Dimensional DPCM System. This is because in KL-Cosine-DPCM System a bank of DPCM modulators are used where the KL-2 Dimensional System uses a single DPCM modulator. Using a number of DPCM modulators increases the total number of output symbols, thus reducing the number of times each symbol occurs. This will reduce the effectiveness of the variable-length encoder in reducing the output bit rate. The end result is that the performance of the KL-2 Dimensional DPCM System improves an average of 25% where the performance of the KL-Cosine-DPCM System improves by about 10%.

Comparison of the Recommended Techniques Based on Recognition Accuracy

Figure 5 shows the performances of the three recommended bandwidth compression methods in "preserving the classification accuracy of the reconstructed multispectral imagery." Using this criterion the comparative performance of the KL-2 Dimensional DPCM System and KL-Cosine-DPCM System is basically the same as it was for using signal-to-noise ratio as the criteria of optimality. This shows a strong correlation between the signal-to-noise ratio and the classification accuracy.

Figure 5 also shows the performance of the cluster coding method. The classification accuracy decreases as a result of increasing the number of clusters in each block up to 16 clusters per block. Then the classification accuracy increases sharply when 32 clusters per block is allowed. This behavior is due to the fact that the cluster coding algorithm is using a classification procedure on blocks of 16 by 16 samples while classification accuracy is measured using a classification procedure on blocks of 256 by 256 samples. A classification accuracy of 100% results if the block size for measuring the classification accuracy and the block size in cluster coding methods are the same. The argument for using large block sizes in classifying multispectral data is not very solid. This depends very much on users and applications of multispectral imagery. Large block sizes are likely to be used in applications where one is only interested in a limited number of objects in a large segment of imagery, as in hydrology and urban planning. For this application the imagery obtained using cluster coding algorithm has a rather poor classification accuracy compared to the other two recommended bandwidth compression algorithms.

Comparison of the Recommended Techniques Based on System Complexity

The implementational complexity of the selected bandwidth compression techniques is summarized in Table 1. The results are in terms of the number of computations and memory per picture element. This table also shows typical numbers for 4 channel data and the block sizes used in simulating the compression methods. A detailed analysis of the complexity of the cluster coding method is not performed. But based on the computer time required for compressing the bandwidth of the representative LANDSAT data using this system and comparing it with the computer time required in simulating the KL-2 Dimensional DPCM encoder, it was concluded that the cluster coding algorithm is about one order of magnitude more complicated.

Comparison of the Recommended Techniques Based on Channel Noise and Sensor Effects

Figure 6 shows the reduction in the signal-to-noise ratio and the classification accuracy which is caused by introduction of a noisy channel. The results refer to a binary symmetric channel with bit-error rates ranging from 10^{-4} to 10^{-2} . Both algorithms degrade significantly at higher bit-error rates. The system using the KL-2 Dimensional encoder has greater degradation. Figure 7 shows the reconstructed form of band 3 of the LANDSAT data after bandwidth compression using Haar - 2 Dimensional DPCM encoder at bit error rates of 10^{-3} and 10^{-4} . The propagation of the channel error is clearly visible for bit-error rate of 10^{-3} . However, this is less significant than the effect of channel error for a simple two-dimensional DPCM system. This is because the recommended system using Haar - 2 Dimensional DPCM encoder utilizes the two-dimensional DPCM encoder in the spatial domain. The signal at the receiver of the two-dimensional DPCM system is then transformed by the inverse of the Haar transform to reconstruct multispectral data. The total effect of the inverse Haar transform at the receiver is to distribute the channel error (and its propagation) among all spectral bands. Although this leaves the total error unchanged, it distributes the channel error among all spectral bands thus making it less objectionable to the human vision.

The sensor imperfections affect the correlation of the data which in turn affects the performance of the bandwidth compression techniques. Analytical results show that this effect is minimal.

Conclusions

The following are the major conclusions of the study

- (1) Fixed rate encoders give excellent quality for the reconstructed imagery at 4 to 1 compression ratio. This compression ratio is increased to 6 to 1 by making the system operate at a variable rate by adding a Huffman encoder to the system. At this compression ratio the method using Haar 2 Dimensional DPCM system has a signal-to-noise ratio better than 35dB, a classification accuracy better than 91% and gives reconstructed imagery which are almost indistinguishable from the originals.
- (2) At a fixed bit rate and a compression ratio of 8 to 1 the Haar-Cosine-DPCM system gives acceptable results. This corresponds to signal-to-noise ratio of better than 31dB, recognition accuracy of better than 82% and slightly degraded imagery.
- (3) Cluster coding methods can be used to obtain a compression ratio of 12 to 1 which corresponds to a signal-to-noise ratio of better than 29dB, recognition accuracy of larger than 71% and possibly acceptable reconstructed imagery.
- (4) Compression ratios of 16 to 1 and higher correspond to noticeably degraded imagery which are only acceptable for some users.
- (5) The effect of the sensor imperfection on the performance of the candidate bandwidth compression methods was minimal. Therefore, it had minimal impact on the choice of the recommended techniques.
- (6) Channel error effects are minimal for transform and DPCM systems for bit error rates less than 10^{-4} .

References

- [1] Definition of the Total Earth Resources System for the Shuttle Era, General Electric Space Division, Contract No. NAS9-13401, NASA Lyndon B. Johnson Space Center, Houston, Texas 77058.
- [2] Multispectral Scanner Data Redundance Study, Final Report, May 15, 1971, Philco-Ford, Contract No. NAS5-21162, NASA/Goddard Space Field Center, Greenbelt, Maryland.
- [3] P. J. Ready, P. A. Wintz, "Information Extraction, SNR Improvement, and Data Compression in Multispectral Imagery", IEEE Transactions on Communications, Vol. 21, October 1973, pp. 1123-1131.
- [4] J. N. Gupta, P. A. Wintz, "A Boundary Finding Algorithm and its Applications", IEEE Transactions on Circuits and Systems, Vol. CAS-22, No. 4, April 1975, pp. 351-362.
- [5] C. L. May, D. J. Spencer, "Data Compression for Earth Resources Satellites", International Telemetry Conference, Los Angeles, September 1972.
- [6] E. E. Hilbert, Joint Classification and Data Compression of Multidimensional Information Source Application to ERTS, Conference Record Vol. II, International Conference on Communications, June 1975, pp. 27(1-6).
- [7] A. Habibi, "Two-Dimensional Bayesian Estimate of Images", IEEE Proc. Vol., 60, No. 7, July 1972 pp. 878-883.
- [8] A. Habibi, "Delta Modulation and DPCM Coding of Color Signals", Proceedings of International Telemetry Conference, Dec. 1972, pp. 363-373.
- [9] A. Habibi, "Hybrid Coding of Pictorial Data", IEEE Trans. on Communications, Vol. COM-22, No. 5, May 1974, pp. 614-624.
- [10] J. A. Roese, W. K. Pratt, G. S. Robinson, A. Habibi, "Interframe Transform Coding and Predictive Coding Methods", Conference Record of International Conference on Communications, June 1975, pp. 23(17-21).

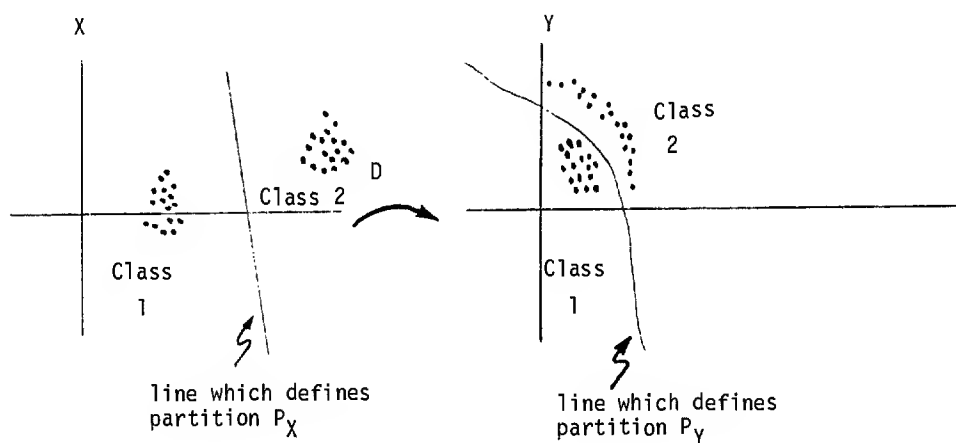


Figure 1. Partitioning Data in Two Classes Before and After Distorting the Data.

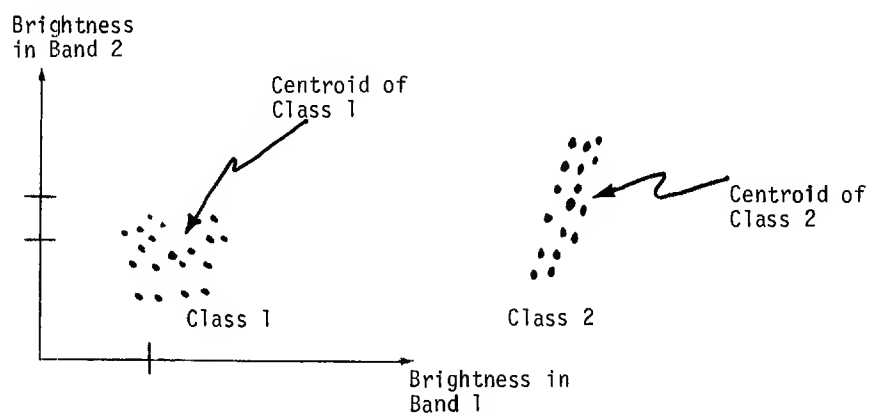


Figure 2. Each Class has a Centroid Equal to the Average Location of Data in that Class.

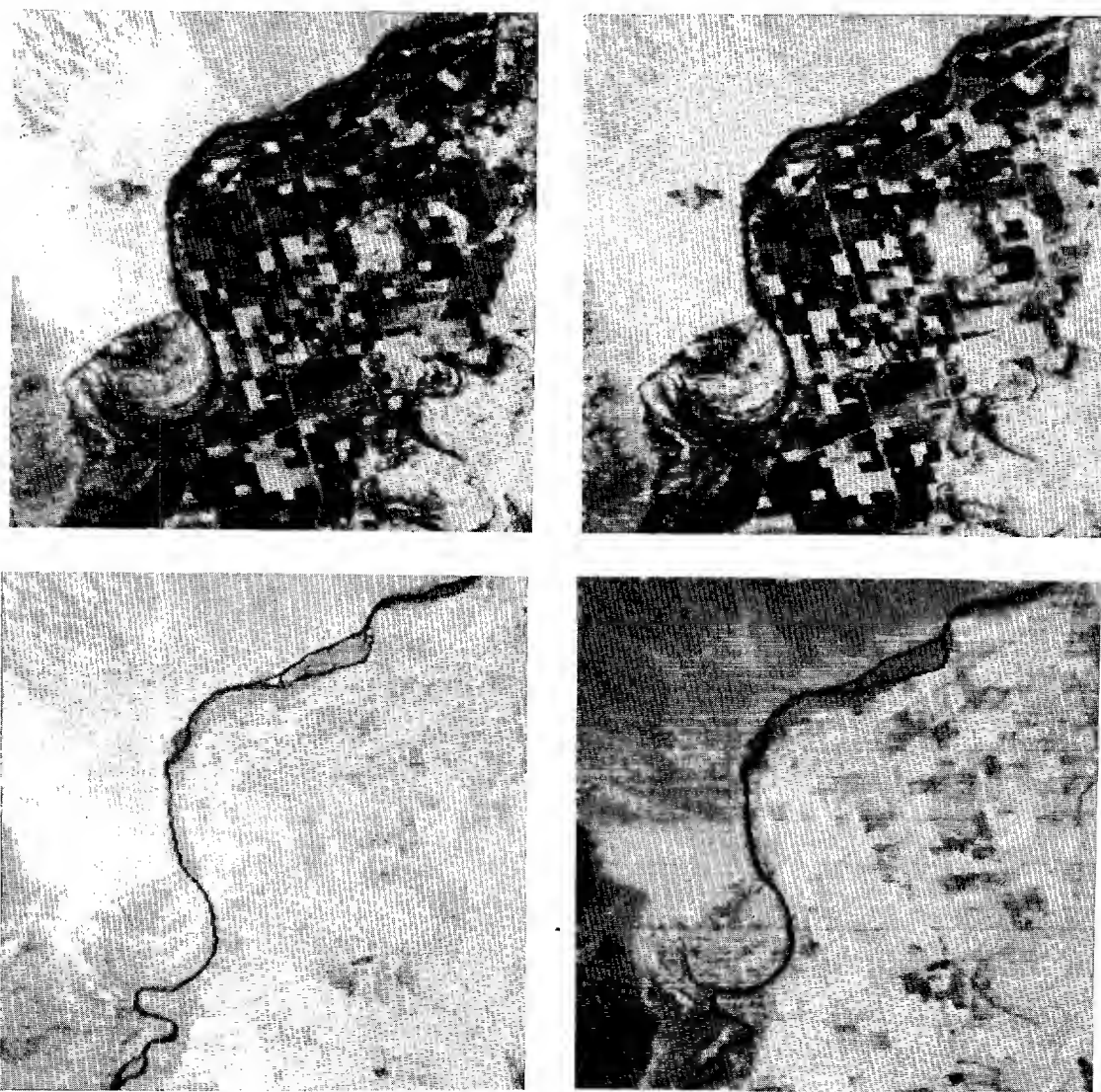


Figure 3. Original Bands

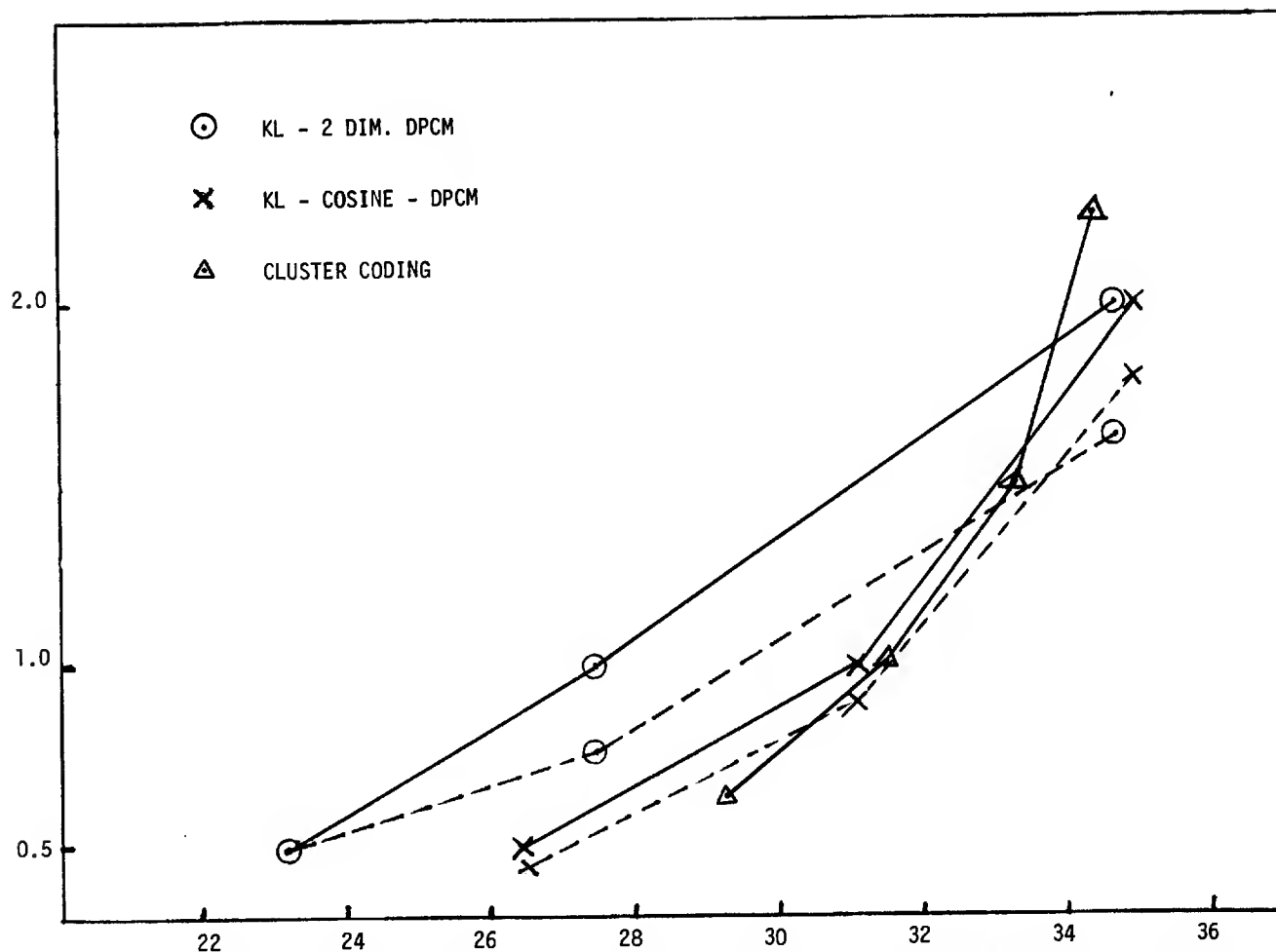


Figure 4. Bit Rate Versus Signal-To-Noise Ratio for KL-2Dim, DPCM, KL-Cosine-DPCM, and Cluster Coding.

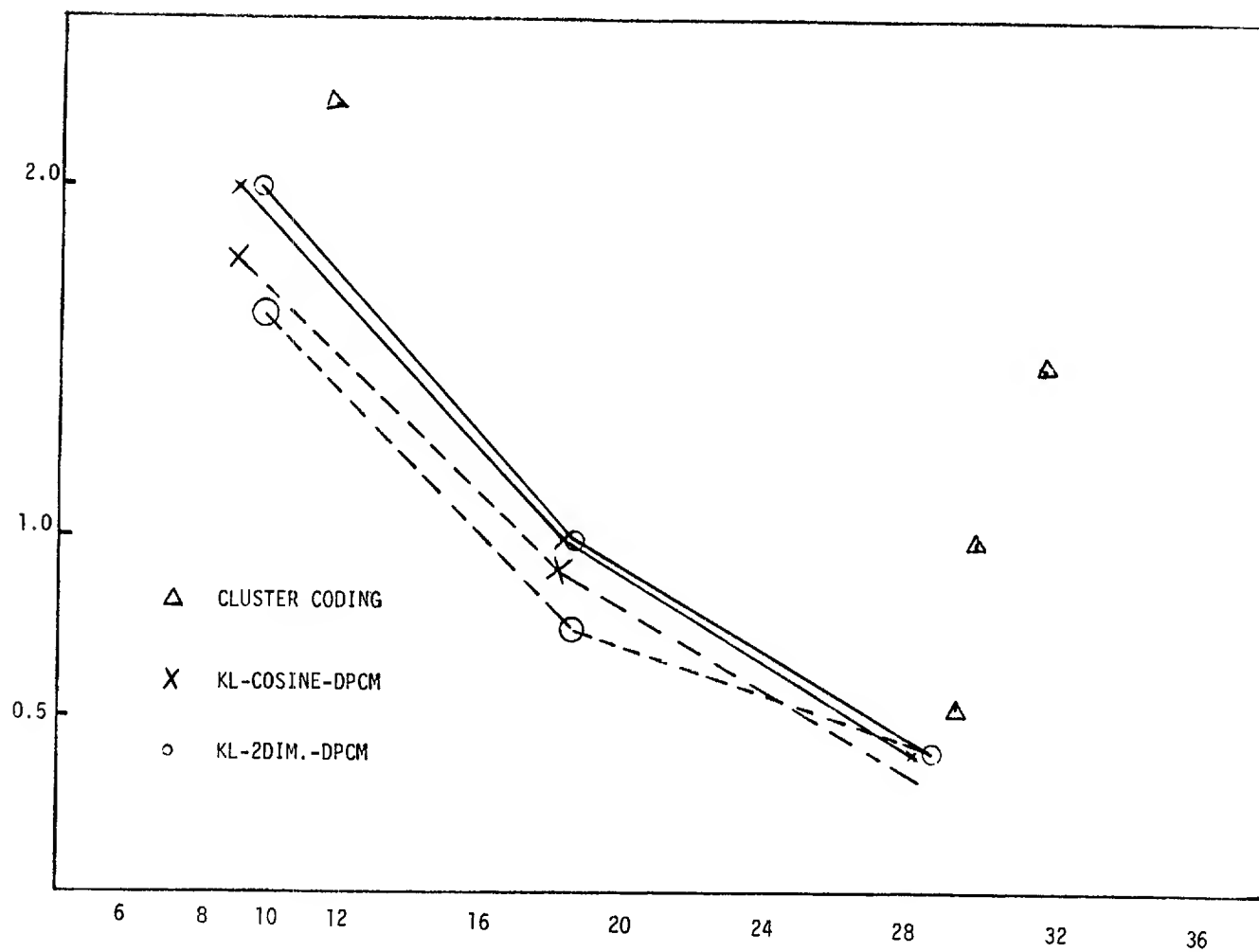


Figure 5. Bit Rate Versus Classification Inaccuracy

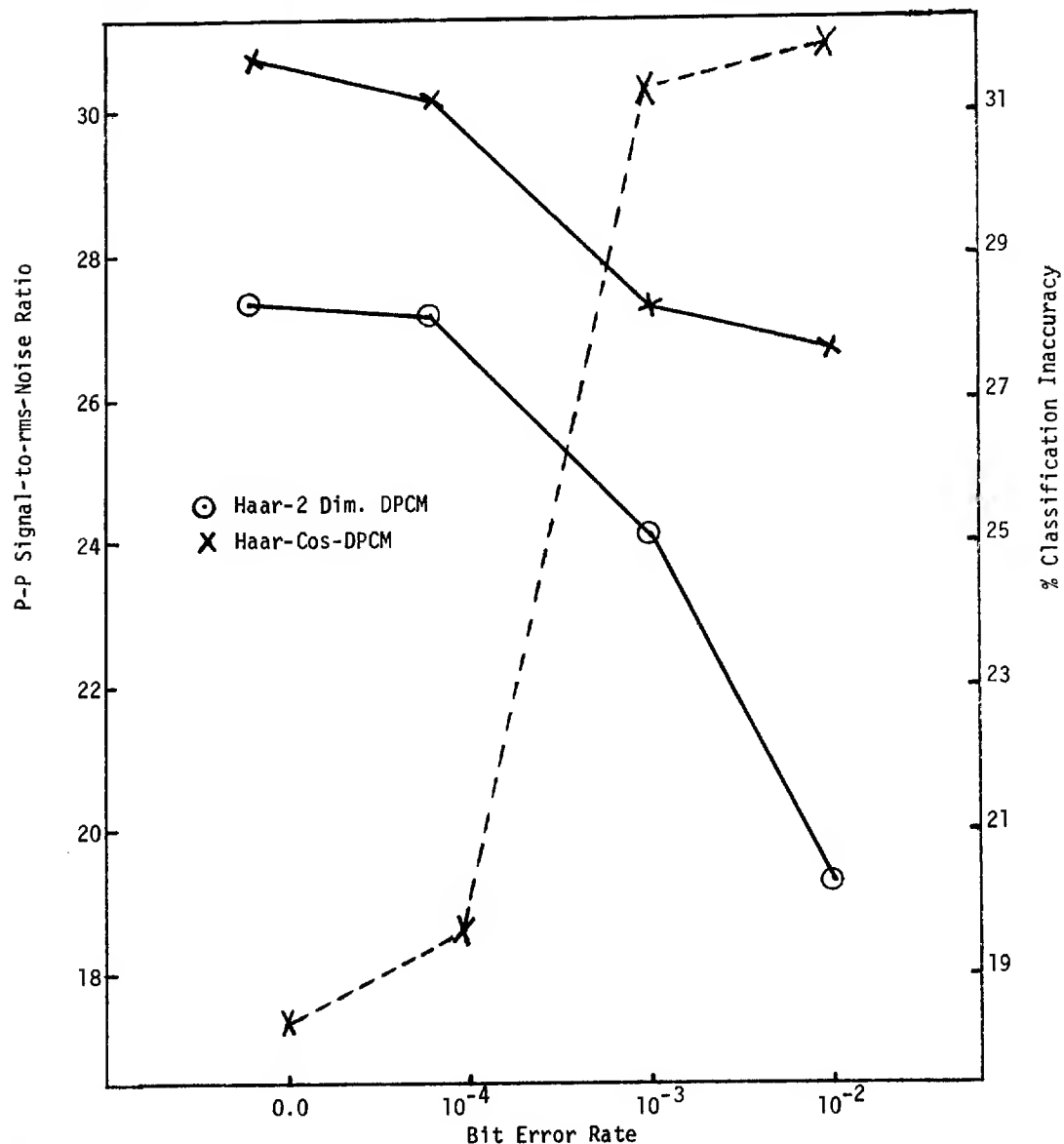
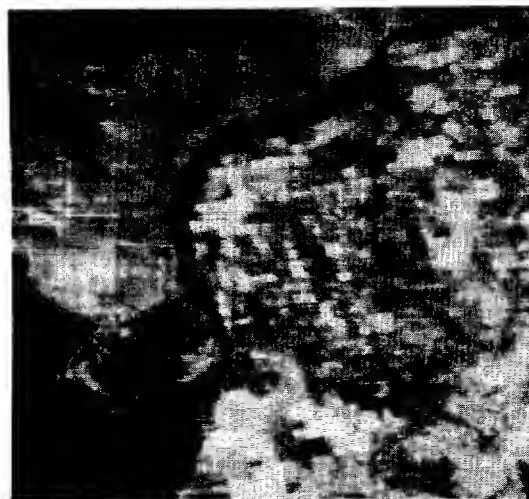


Figure 6. P-P Signal-to-Noise and Classification Inaccuracy (Dash line) versus Bit Error Rate.



$P = 10^{-4}$



$P = 10^{-3}$

Figure 7. Band 3 - 1 Bit - Haar - 2 Dim. DPCM.

Table 1. Computational Complexity of 3 Bandwidth Compression Methods

	COMPUTATIONAL NEEDS			STORAGE NEEDS (BITS)	
	ADDS (OR SUBTRACTS)/ PIXEL	MULTIPLIES/ PIXEL	RECORD STORAGE	MEMORY	
3-Dimensional <u>Hadamard Transform</u> and Block Quantization	$\log_2 B^2 S + T$	1	16BSW	$44B^2 S + 8000$	
	TYPICAL	TYPICAL	TYPICAL	TYPICAL	TYPICAL
1-Dimensional <u>Karhunen-Loeve Transform</u> on the spectral domain followed by 2-Dimensional <u>DPCM</u>	$S + T + 3$	$S + 4$	16SW	$16S^2 + 148S + 8000$	
	TYPICAL	TYPICAL	TYPICAL	TYPICAL	TYPICAL
Karhunen-Loeve Transform on the spectral domain followed by a Had- amard Transform in the scan direction (x-direction) with y-direction DPCM using block quantization.	$\log_2 B + S + T + 1$	$S + 2$	32SW	$84BS + 32B + 32S(S + 1) + 8000$	
	TYPICAL	TYPICAL	TYPICAL	TYPICAL	TYPICAL

TYPICAL

B; Block size
 T; Bits/pixel
 S; Spectral Bands
 W; Sample in each scan

B = 16
 T = 1
 S = 4
 W = 3200

REAL TIME TELEVISION IMAGE BANDWIDTH REDUCTION USING CHARGE TRANSFER DEVICES

H. J. Whitehouse
R. W. Means
E. H. Wrench

Abstract

Recent advances in analog semiconductor technology have made possible the direct sensing and processing of television images. By combining a charge transfer device (CTD) imager and a CTD transversal filter, real time image sensing and encoding have been achieved with low power integrated circuits so that digital transmission and bit rate reduction are made possible using differential pulse code modulation (DPCM). Good mean square error performance and freedom from DPCM artifacts are possible in a hybrid intraframe image encoder. The hybrid transform encoder performs a discrete cosine transform (DCT) on each line of the television image as it is scanned. This compacts the variance into low frequency coefficients and the DPCM encodes the corresponding DCT coefficients between successive lines. Computer simulation of this hybrid coding technique has shown good performance on 256 x 256 pixel images at 0.5 bits/pixel and channel bit error rates of 10^{-2} . An experimental system using a low resolution General Electric 100 x 100 charge injection device camera and a Texas Instruments bucket brigade transversal filter as part of the DCT processor has been constructed and provides good low resolution image quality at 1 bit/pixel and bit error rates of 10^{-3} . A high resolution vidicon compatible system is also being constructed.

Introduction

Unitary transforms for image encoding have been used for intraframe encoding.⁽¹⁾ In addition, these techniques may also be applied to interframe and multispectral encoding. However, all unitary transformations are information preserving and no bandwidth reduction results from the application of the transform to the image. Instead, the transforms redistribute the variance associated with each picture element (pixel) so that subsequent to the transform, basis restricting operations on the transform coefficients will result in bandwidth reduction. Upon reconstruction of the original image from the basis restricted transform coefficients, a degraded version of the original image can be obtained. Unfortunately, the interrelationship between the type of transform, the form of the noninvertible operation, and the type of degradation in the reconstructed image is very complicated and subjective. The universally used analytic criterion of the mean-square-error is, at present, the best compromise technique for transform comparison.

For the particular operation of basis restriction by truncation, a particularly simple interpretation of the bandwidth reduction can be made. The transforms may be viewed as a variance redistributing operation that approximately decorrelates the transform coefficients while transforming the variance associated with each picture element into the low-order coefficients of the transform. Under the assumption that each set of picture elements can be considered as a sample function from a wide sense stationary random process with correlation function $r|\tau|$, there exists an optimum discrete transformation, the Karhunen-Loeve transformation, which totally decorrelates the transform coefficients and maximally compacts the variance to the low-order coefficients. All other transformations can be compared in their performance by comparing their transform coefficient decorrelation and variance compaction with this optimum transformation.

This intuitive interpretation can be made rigorous through the use of the rate-distortion criterion.^(2, 3) It has been found from experience that the closer the eigenvectors of the transformation are to the eigenvectors of the optimum Karhunen-Loeve transformation the greater the variance is compacted and the more the coefficients can be truncated while maintaining a fixed rate distortion or mean-square-error.

Transform Encoding

Karhunen-Loeve Transformation

If a continuous time function of zero mean and autocorrelation function $R(\tau) = e^{-\alpha|\tau|}$ is considered to be a sample function from a wide-sense stationary random process, then this time function can be explicitly expanded by the Karhunen-Loeve expansion⁽⁴⁾ and the resulting coefficients will be uncorrelated. For a discrete function of zero mean and autocorrelation function $R(\tau) = r|\tau|$, which may be considered as a sample function from a first-order Markov process, a similar discrete Karhunen-Loeve transformation may be defined.⁽⁵⁾ This transformation diagonalizes the covariance matrix and is optimal in the mean-square-error sense for a restricted set of basic functions that do not span the complete space.

The discrete Karhunen-Loeve expansion is given by⁽⁵⁾ for the case $N = 2m$ as

$$G_k = \sum_{n=1}^{2m} \frac{2}{2m + \lambda_n^2} \sin \left\{ \omega_n [k - (2m + 1)/2] + n\pi/2 \right\} g_n$$

$$k = 1, 2, \dots, 2m \quad (1)$$

where

$$\lambda_n^2 = \frac{1 - r^2}{1 - 2r \cos \omega_n + r^2} \quad (2)$$

and ω_n are the first N positive roots of

$$\tan 2m\omega = \frac{-(1-r^2) \sin \omega}{(\cos \omega - 2r + r^2 \cos \omega)} \quad (3)$$

Since the discrete Karhunen-Loeve expansion involves both the solution of a transcendental equation and the evaluation of the autocorrelation function of the data to be transformed, real time computation of this transform is quite complex. However, Habibi and Wintz⁽¹⁾ have shown that Karhunen-Loeve transformations calculated using approximate autocorrelation functions are satisfactory for many applications.

Discrete Fourier Transform

Since the discrete Fourier transform is asymptotic to the Karhunen-Loeve transformation⁽⁶⁾ for the exponential covariance function and the basis vectors are picture independent, the Fourier transform represents a logical choice for real time implementation. The Fourier transform exists for all data lengths N . This is defined by

$$G_k = \sum_{n=0}^{N-1} e^{-i\pi 2nk/N} g_n \quad k = 0, 1, \dots, N-1 \quad (4)$$

Discrete Cosine Transform

Two different types of discrete cosine transform (DCT) are useful for reduced redundancy television image transmission. Both are obtained by extending the length N data block to have even symmetry, taking the discrete Fourier transform (DFT) of the extended data block, and saving N terms of the resulting DFT. Since the DFT of a real even sequence is a real even sequence, either DCT is its own inverse if a normalized DFT is used.

The "Odd DCT" (ODCT) extends the length N data block to length $2N-1$, with the middle point of the extended block as a center of even symmetry. The "Even DCT" (EDCT) extends the length N data block to length $2N$, with a center of even symmetry located between the two points nearest the middle. For example, the odd length extension of the sequence $A B C$ is $C B A B C$, and even length is $C B A A B C$. In both cases, the symmetrization eliminates the jumps in the periodic extension of the data block which would occur if one edge of the data block had a high value and the other edge had a low value; in effect, it performs a sort of smoothing operation with no loss of information. It will be noted that the terms "odd" and "even" in ODCT and EDCT refer only to the extended data block — in both cases the extended data block has even symmetry.

Let the data sequence be g_0, g_1, \dots, g_{N-1} . The ODCT of g is defined as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n e^{\frac{-i2\pi nk}{2N-1}} \quad \text{for } k = 0, 1, \dots, N-1 \quad (5)$$

where

$$g_{-n} = g_n \quad \text{for } n = 0, 1, \dots, N-1. \quad (6)$$

By straightforward substitution it may be shown that

$$G_k = 2 \operatorname{Re} \sum_{n=0}^{N-1} \tilde{g}_n e^{\frac{-i2\pi nk}{2N-1}} \quad (7)$$

where \tilde{g} is defined by equation (8).

$$\tilde{g}_n = \begin{cases} 0.5 g_0, & n = 0 \\ g_n, & n = 1, \dots, N-1 \end{cases} \quad (8)$$

The EDCT of g is defined by equation (9), where the extended sequence is defined by equation (10).

$$G_k = e^{\frac{-i\pi k}{2N}} \sum_{n=-N}^{N-1} g_n e^{\frac{-i2\pi nk}{2N}} \quad \text{for } k = 0, 1, \dots, N-1 \quad (9)$$

$$g_{-1-n} = g_n \quad \text{for } n = 0, 1, \dots, N-1 \quad (10)$$

If the mutually complex conjugate terms in equation (9) are combined, then equation (11) results. Equation (11) may be viewed as an alternate way of defining the EDCT.

$$G_k = 2 \operatorname{Re} \left\{ e^{\frac{-i\pi k}{2N}} \sum_{n=0}^{N-1} g_n e^{\frac{-i2\pi nk}{2N}} \right\}$$

$$= 2 \sum_{n=0}^{N-1} g_n \cos \left[\frac{2\pi (n + 0.5) k}{2N} \right] \quad (11)$$

Ahmed⁽⁷⁾ has investigated the use of the EDCT as a substitute for the Karhunen-Loeve transform and finds that it is superior to the Fourier transform and is comparable to the Karhunen-Loeve (K-L) in rate-distortion performance while maintaining the computation simplicity of a transform which does not depend on the picture statistics. Habibi⁽⁸⁾ has shown by simulation that the DCT is equivalent in a mean-square-error sense to the K-L transform under basis restriction.

Hybrid Transforms

Mixed transforms can be used either for intraframe encoding or interframe encoding depending on the available memory and the type of transforms implemented. In order for a mixed transform to be competitive with one of the conventional two-dimensional transforms, it must offer either superior performance or simplicity of implementation. Of the transforms examined the odd length cosine transform is competitive in performance since it can be implemented as the real part of a CZT and since the transform samples are real and are the samples of an autocorrelation function which may then be extrapolated by well-known techniques. In simulations of transform performance, the cosine transform has been shown to closely approximate the behavior of the Karhunen-Loeve transform.

The benefits of mixed transformation implementation and minimum memory may be achieved for digital transmission by combining a one- or two-dimensional unitary transform with generalized DPCM in a hybrid system. The basis of operation of the hybrid transform is that the unitary transform decorrelates the image within its constraints of transform type, dimensionality, and block size, while the generalized DPCM removes the correlation between transform blocks. This hybrid system is particularly attractive for remote sensor application since it has been found that its performance is approximately as good as the Karhunen-Loeve transform and its implementation requires minimum memory.⁽⁸⁾

Implementation

Computational Modules

A linear transform on sampled data of finite extent may be viewed as the multiplication of a vector by a matrix. Multiplication by diagonal, circulant, or Toeplitz matrices may be accomplished rapidly with simple computational hardware modules. Multiplication by an $N \times N$ diagonal matrix requires only a scalar multiplier and a memory containing N values to provide serial access to the reference function. Multiplication by an $N \times N$ Toeplitz matrix corresponds to a convolution and may be performed using a transversal filter having $2N-1$ taps. Multiplication by an $N \times N$ circulant is a special case of multiplication by $N \times N$ Toeplitz matrix in which the length of the transversal filter may be reduced to N taps if the data block is recirculated through the filter or reread into the filter from a buffer memory.

One-Dimensional DFT

Linear filters have been used for many years for the calculation of the power spectra of continuous signals. One of the earliest methods used a bank of wave filters to measure the spectra in fractional octave bands for telephone network equalization.⁽⁹⁾ However, when increased resolution was required the number of filters rapidly become unmanageable. An alternative which overcame the difficulty of a large number of filters each with small time-bandwidth product was to substitute one linear fm (chirp) filter with large time-bandwidth product and to employ matched filtering. In this system the signal to be analyzed is used to single sideband (SSB) modulate a locally generated chirp signal and the composite modulated signal is filtered in a chirp delay line matched filter. Each component of the input signal spectrum shifts the locally generated chirp to a different position in the spectrum after SSB modulation and these shifted chirps then correlate with the reference signal represented as the impulse response of the matched filter at different times. Thus the output signal amplitude-time history reflects the amplitude-frequency composition of the input signal.

Bleustein⁽¹⁰⁾ recognized that the discrete Fourier transform (DFT) of sampled data was amenable to a similar interpretation. In addition to just calculating the magnitude of the Fourier transform, linear filters could calculate the phase and thus all of the operations such a cross convolution and a cross correlation could be calculated. This technique came to be called the chirp-Z transform (CZT) and can be applied to other problems besides just the calculation of the DFT.⁽¹¹⁾ Prior to these developments, digital computation of the DFT had been significantly improved by the use of a special algorithm called the fast Fourier transform (FFT) which was described by Cooley and Turkey.⁽¹²⁾ The FFT algorithm gained rapid popularity in signal processing since it allowed the calculation of the DFT to be done using significantly fewer machine operations (multiplications) than direct evaluation.

By direct inspection it is observed that, if symmetries of the function $\exp j\pi 2nm/N$ are not exploited, then the number of complex multiplications required will be N^2 corresponding to N multiplications for each frequency component evaluated. Even on high speed digital computers this can become the limiting consideration in signal processing applications. The advantage of the FFT algorithm is that for highly composite values of the DFT size N the number of multiplications is proportional to $N \log_2 N$.

Although the FFT has been successful in substantially reducing the computing time and cost of using general purpose digital computers it has several disadvantages for special purpose real time computation. At high throughput rates which are required for real time image processing the processor either must operate $\log_2 N$ times faster than the data rate or pipeline structures which use distributed

memory and $\log_2 N$ multipliers must be used. In addition, the internal arithmetic of the FFT processor must be done at increased precision in order to compensate for the multiple round off errors introduced by the successive stages in the FFT processor. Although these difficulties can be overcome, it is not always possible to arrange the computation in a form where the size of the transform is highly composite. For the above reasons and because of the difficulty of obtaining small, low power, fast analog to digital converters, linear transversal filter implementations of the chirp-Z-transform are attractive⁽¹³⁾ rather than the previous CZT implementation which used an FFT to perform the required convolution.

The DFT may be easily reduced to the form suitable for linear filtering by the substitution

$$2nm = n^2 - (n - m)^2 + m^2 \quad (12)$$

which changes a product of variables into a difference so that

$$G_m = e^{-j\pi m^2/N} \sum_{n=0}^{N-1} e^{j\pi(n-m)^2/N} e^{-j\pi n^2/N} g_n \quad (13)$$

This form is seen to be equivalent to factoring the Fourier matrix F into the product of three matrices

$$F = DTD \quad (14)$$

where D is a diagonal matrix with elements $d_{nn} = \exp(-j\pi n^2/N)$ and T is a Toeplitz matrix with elements $t_{nm} = \exp(j\pi(n-m)^2/N)$.

The CZT algorithm is easily implemented by transversal filter techniques. In this case the DFT is computed by premultiplication by a discrete chirp, convolution with a discrete chirp, and postmultiplication by a discrete chirp. Figure 1 shows this configuration. However, it must be remembered that both the multiplications and convolutions are complex and a suitable representation of the complex numbers must be used. One representation is by real and imaginary part. Figure 2 shows the DFT organized as a CZT and implemented with parallel computation of the real and imaginary parts. In Figure 2 the input signal is represented as $g = g_R + jg_I$ and the output signal is represented as $G = G_R + jG_I$, where it is understood that $g = g_n$ $n = 0, \dots, N-1$ and $G = G_n$ $n = 0, \dots, N-1$.

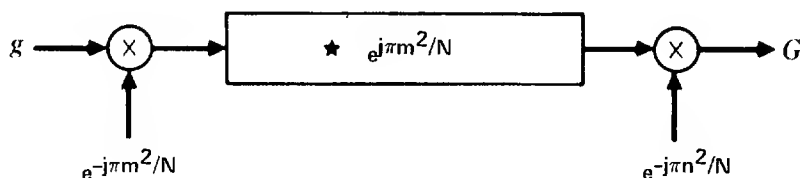
In order to determine the specific form of the transversal filters it is necessary to know the specific value of N . When N is odd the Toeplitz matrix T may be represented as a transversal filter with $2N-1$ complex taps $h_{-(N-1)}$ to h_{N-1} where $h_n = W^{-n^2/2}$, $n = -(N-1)$ to $N-1$, and $W = \exp(-j2\pi/N)$. The required convolution has been implemented with the general transversal filter shown in Figure 3.

When N is even, it can be shown that $T_{n,m} = T_{N+n,m}$ where the subscripts are reduced mod N . Thus T is a circulant matrix and can be implemented with a recirculating transversal filter as shown in Figure 4 where the number of complex taps is N and the tap weights are: $h_n = W^{-n^2/2}$ $n = 0, \dots, N-1$.

Charge Coupled Devices

CCDs are sampled data analog circuits which can be fabricated by Metal Oxide Semiconductor (MOS) technology as LSI components.⁽¹⁴⁾ As such they are directly compatible with other MOS circuits. Current CCD transversal filters have operated as video devices with sample rates up to 5 MHz. CCDs operate by the manipulation of injected minority carriers in potential wells under MOS capacitors and thus behave as capacitive reactances with low power dissipation. However, since the potential wells which contain the minority carriers also attract thermally generated minority carriers, there is a maximum storage time for the analog signal which depends on the dark current associated with the temperature of the silicon. Under normal conditions at room temperature, dark currents are tens of nAmps/cm² and storage times of hundreds of milliseconds can be achieved.

There are many ways in which unidirectional charge transfer can be achieved. The first developed was a three-phase clocking structure which is illustrated in the transversal filter of Figure 5. The three electrode CCD structure is planar, much like the SAW devices, and the direction of charge propagation is determined by the sequence of potentials applied to the three electrodes. Unfortunately, if the minority carriers are allowed to collect at the semiconductor-oxide boundary, poor charge transfer efficiency will result due to minority carriers getting caught in trapping sites. This means that the CCD will behave nonlinearly unless there is sufficient propagating charge present to fill all of the traps. By biasing the operating condition of the CCD so that about 10% of the dynamic range is used for the injection of a "fat zero," the traps are kept continuously filled and the device has over a 60 dB dynamic range. In practice, a video signal representing the signal to be processed is added to a fixed bias somewhat larger than one-half of the peak-to-peak value of the signal. Since the effective storage time of the device is long relative to the time required to execute a convolution, CCDs can be considered to be



* Denotes either convolution or circular convolution

Figure 1. Chirp-Z Transform Implementation of the DFT

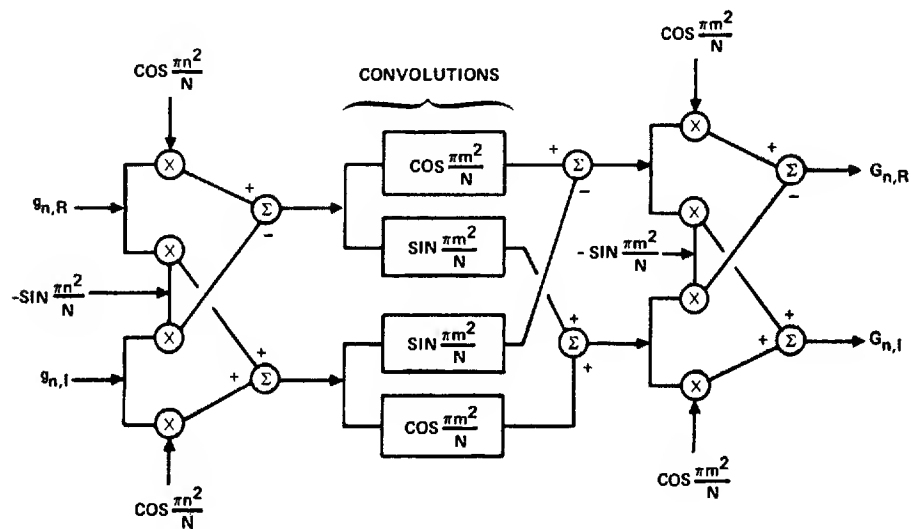


Figure 2. DFT Via CZT Algorithm with Parallel Implementation of Complex Arithmetic

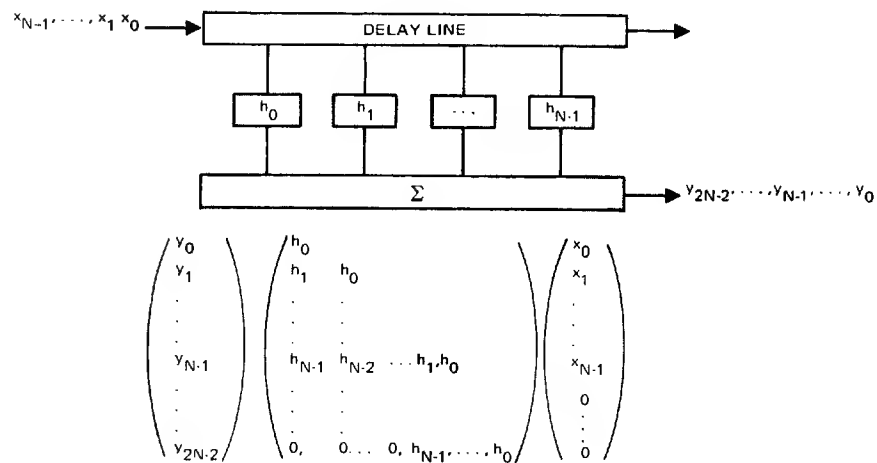


Figure 3. Transversal Filter

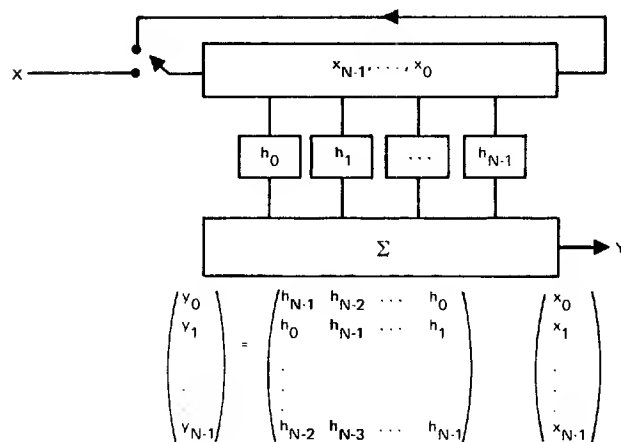


Figure 4. Circular Convolution

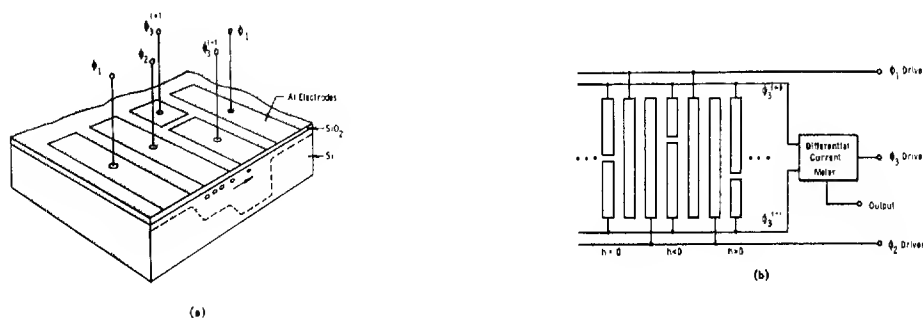


Figure 5. Schematic of the Sampling, Weighting, and Summing Operation

interruptible signal processors and as such are more compatible with the executive control required for signal processing. A 64 point CCD filter with discrete cosine transform sine and cosine chirps is shown in Figure 6. This chip was developed by Texas Instruments for the Naval Undersea Center for image processing. In addition to the four DCT filters, four DFT filters, a Hilbert transform and other experimental signal processing functions were also implemented.

Current research in CCDs is directed toward improving the charge transfer efficiency and removing the requirement of continuous "fat zero" charge injection by ion implantation techniques which keep the minority carriers away from the semiconductor oxide boundary. Ion implantation is also being used to provide asymmetric potential wells so that simpler two-phase clocking can be employed. Currently available CCDs have 500 stages with 0.9999 transfer efficiency and devices with up to 2000 stages are planned.

Another charge transfer device similar to the CCD is the Bucket Brigade Device (BBD). This is a sequence of MOS transistors coupled together by diffusion enhanced Miller capacitance. Although these devices do not operate at frequencies as high as CCDs, they have better low frequency performance since they include active devices. A CZT has been implemented with two BBD chips. Two 200 tap filters are implemented on each chip: one a discrete cosine and the other a discrete sinc filter. The BBD chip is shown in Figure 7. The complex chirp used in the premultiplier and a typical input and output are shown in Figure 8. The input is an offset cosine wave and the output shows a D.C. component plus a response at the cosine wave frequency. These filters can operate at 100 kHz and have tap accuracies better than 1%. With careful control of geometry, both BBD and CCD filters with tap accuracies approaching 0.1% should be possible. This chip was also developed by Texas Instruments for the Naval Undersea Center.

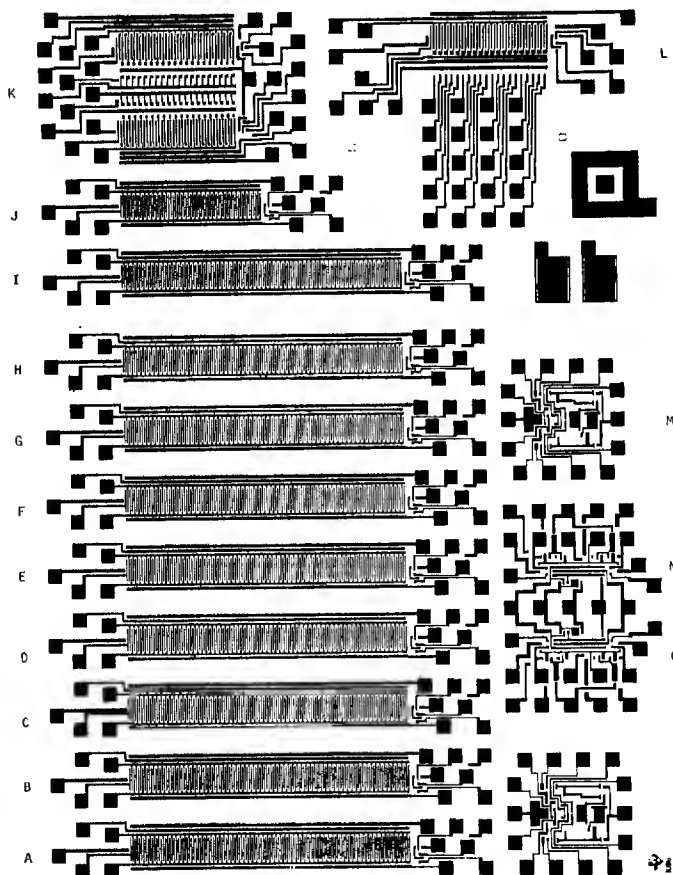


Figure 6. 64 Point CCD Filters

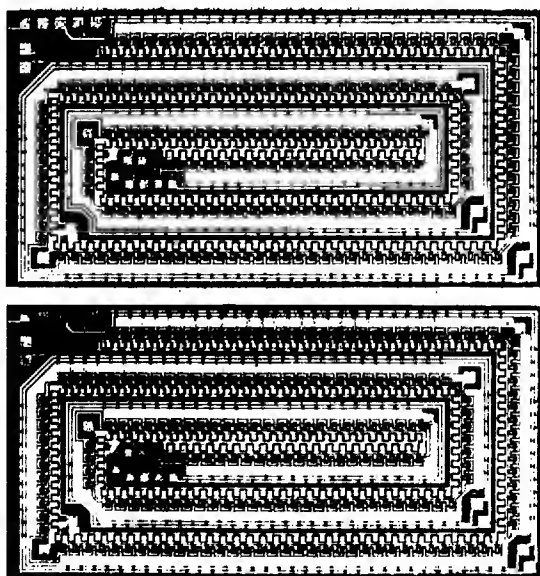
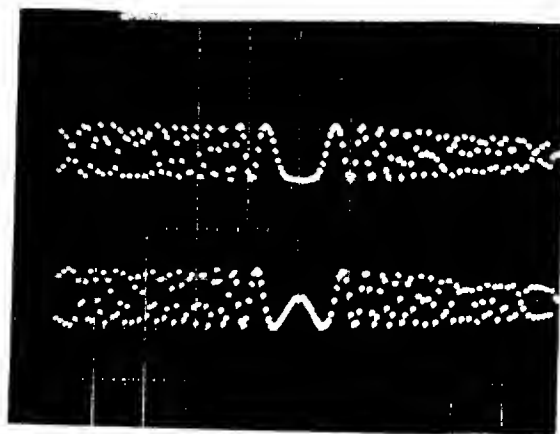
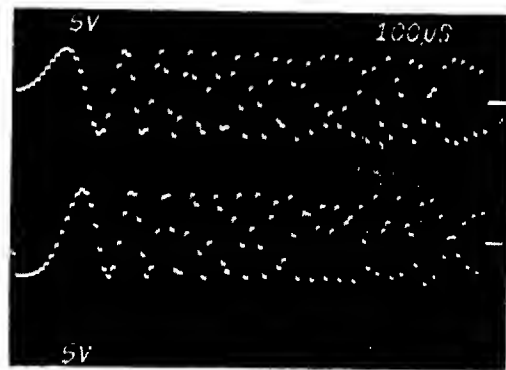


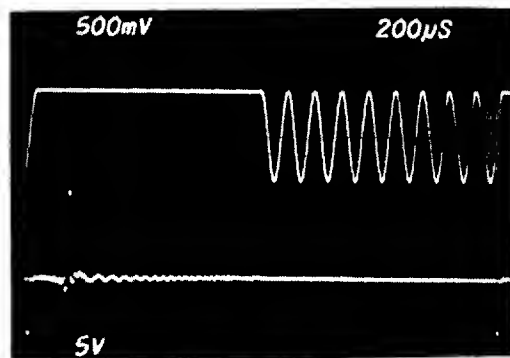
Figure 7. Bucket Brigade Chirp Filters



IMPULSE RESPONSE



PREMULTIPLIER CHIRP



INPUT TO DCT

DCT OUTPUT

Figure 8. BBD Performance

Discrete Cosine Transform

Let the data sequence be g_0, g_1, \dots, g_{N-1} . The ODCT of g is defined in (5) as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n e^{\frac{-j2\pi nk}{2N-1}} \text{ for } k = 0, 1, \dots, N-1 \quad (5)$$

The identity (15) may be used to obtain the CZT form of the ODCT shown in equation (16).

$$2nk = n^2 + k^2 - (n-k)^2 \quad (15)$$

$$G_k = 2 \operatorname{Re} \left\{ e^{\frac{-j\pi k^2}{2N-1}} \sum_{n=0}^{N-1} e^{\frac{-j\pi n^2}{2N-1}} g_n e^{\frac{j\pi (n-k)^2}{2N-1}} \right\} \quad (16)$$

The block diagram of the ODCT is shown in Figure 9. Since only real inputs and outputs are required, a simplified implementation is possible and is shown in Figure 10.

A corresponding implementation may be found for the EDCT. The EDCT of g is defined by equation (9), where the extended sequence is defined by equation (10).

$$G_k = e^{\frac{-j\pi k}{2N}} \sum_{n=-N}^{N-1} g_n e^{\frac{-j2\pi nk}{2N}} \text{ for } k = 0, 1, \dots, N-1 \quad (9)$$

$$g_{-1-n} = g_n \text{ for } n = 0, 1, \dots, N-1 \quad (10)$$

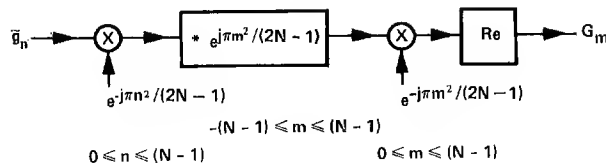


Figure 9. ODCT Block Diagram

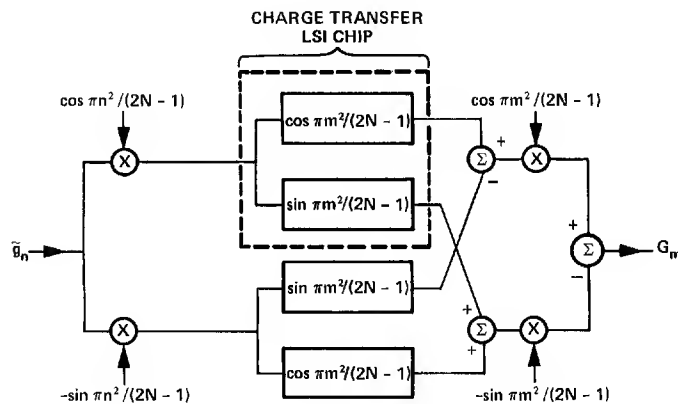


Figure 10. ODCT Expanded Block Diagram

If the mutually complex terms above are combined and the identity (15) used the CZT form of the EDCT becomes

$$G_k = 2 \operatorname{Re} \left\{ e^{\frac{-j\pi k}{2N}} e^{\frac{-j\pi k^2}{2N}} \sum_{n=0}^{N-1} g_n e^{\frac{-j\pi n^2}{2N}} e^{\frac{j\pi(n-k)^2}{2N}} \right\} \quad (17)$$

Simultaneous Computation of the DFT and the DCT

The close relationship between the DFT and the DCT permits the use of common modules to simultaneously compute both transforms. This may be accomplished most simply when an EDCT is computed using DFT modules. The sum in the EDCT defining equation (17) may be interpreted as a length $2N$ DFT of the extension of the function g by N zeros. This leads to the configuration shown in Figure 11. Alternatively, if the odd and even frequencies in the zero-filled DFT are considered separately, they may be computed using length N DFT modules as shown in Figure 12.

System Descriptions

Two hybrid DCT/DPCM bandwidth reduction systems have been selected for construction and evaluation. A block diagram of the systems is shown in Figure 13. The first uses a slow scan image sensor and a Bucket Brigade Device (BBD) transform implementation. The second uses an ordinary vidicon sensor and a Charged Coupled Device (CCD) transform implementation.

In the BBD system a 100×100 pixel solid state sensor is used. The nominal horizontal line scan time is one millisecond. The nominal frame rate is 10 frames per second which can be displayed without flicker through the use of a scan converter. The 1 millisecond line scan time was chosen in order to match the sensor to the BBD filter which operates at a clock rate of 100 kHz with good charge transfer efficiency. At 10 frames per second, image motion should be reproduced well enough for many applications, even though some picture detail is lost because of the low spatial sampling afforded by the 100×100 pixel format. Minimum overall bit rate is achieved by a combination of zonal filtering and variable bit assignment with low spatial frequencies assigned more bits of quantization than high spatial frequencies.⁽¹⁾

The second bandwidth reduction system is compatible with a standard vidicon camera. It uses CCD filters for the cosine transform which will operate at 4.8 MHz sampling rate with a block size of 32 pixels. Compatibility with standard television format is maintained

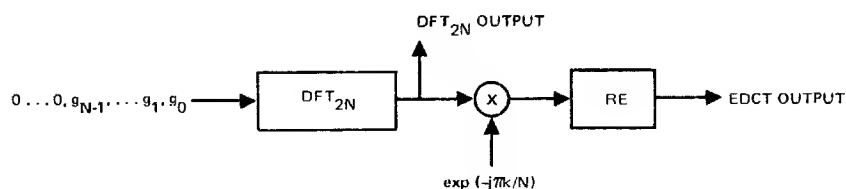


Figure 11. Computation of the DFT and EDCT Using a Single Length $2N$ DFT Module

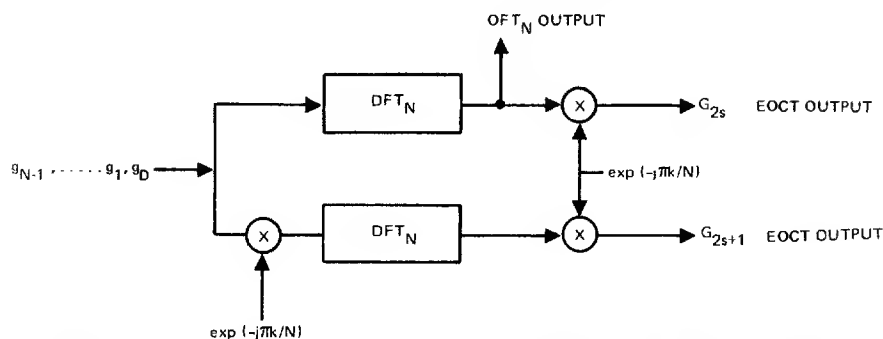


Figure 12. Computation of the DFT and EDCT Using Two Length N DFT Modules

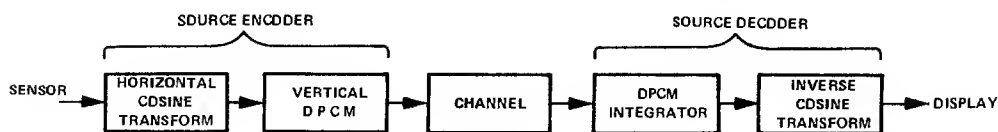


Figure 13. Image Transmission System

in as many aspects as possible. If the interlace field is used directly as the input to the transform hardware, a resolution of approximately 240 lines by 256 pixels is possible at 60 fields/sec. This is equivalent to a video bandwidth slightly less than 2.5 MHz. Computer simulation of this system is shown in Figure 14 for 1 bit/pixel. P is the channel bit error rate.

The implementation of selected transforms via transversal filters was proposed at the All Applications Digital Computer Conference.⁽¹³⁾ The computation of the discrete Fourier transform has been demonstrated using surface wave devices.⁽¹⁵⁾ The discrete cosine transform has been demonstrated using bucket brigade devices.⁽¹⁶⁾

The DCT implementation block diagram is shown in Figure 10. The convolutions are performed in both television systems by charge transfer devices built by Texas Instruments. The multiplications are performed by conventional circuitry. The reference functions used in the pre- and post-multiply can be stored in a real only memory.

Figure 15 is the implementation block diagram for the DPCM part of the system. The memory is a line store which is used as the predictive element in the DPCM. The cosine transform coefficients stored in the memory are subtracted from the new transform coefficients as they enter the DPCM. The difference is quantized and transmitted with a selectable number of bits per coefficient, the number being a function of the assumed variance of the coefficient. The difference coefficients are then added to the previous line coefficients to create the predictive element for the next line.

The BBD slow scan system is shown in Figure 16. It consists of a discrete cosine transform, a DPCM, a channel simulator in which bit errors can be injected, an inverse DPCM, and an inverse DCT. The system is built on eight wire wrapped boards. A blow-up of one is shown in Figure 17. The two chips in the center are the BBD devices. System performance is illustrated in Figure 18 at 2 bits/pixel. Because the original picture has only 100 x 100 pixels, adjacent pixels are not as well correlated as in the 256 x 256 simulation and bandwidth compression algorithms do not work as well. The system performance at one bit per pixel is shown in Figure 19. Results from the CCD high resolution system are not yet available.

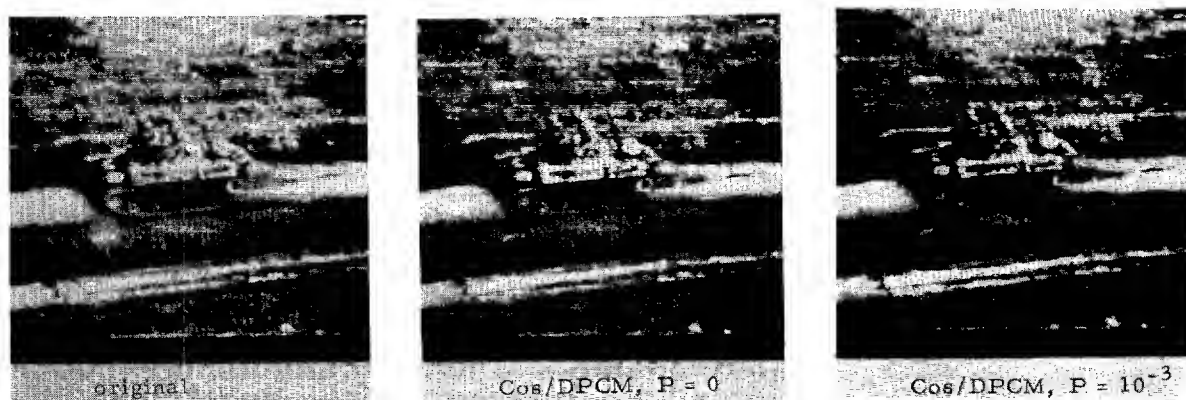


Figure 14. Computer Simulation of CCD System

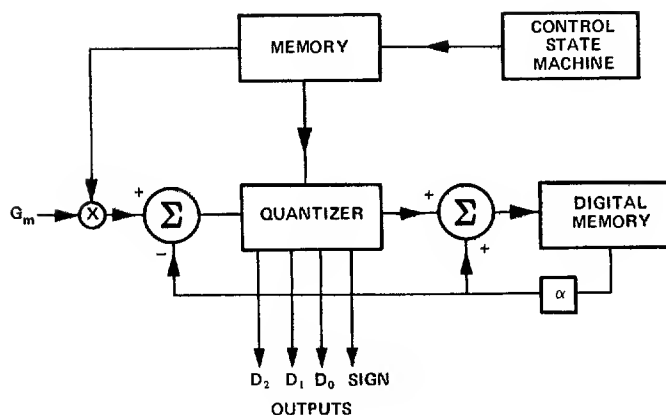


Figure 15. DPCM Encoder

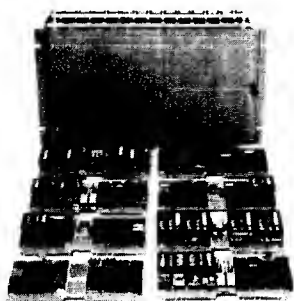


Figure 16. BBD System

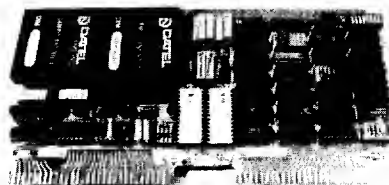


Figure 17. BBD Board



ORIGINAL



2 bits/pixel $P = 0$



2 bits/pixel $P = 10^{-3}$

Figure 18. BBD System Performance



1 bit/pixel $P = 0$



1 bit/pixel $P = 10^{-3}$

Figure 19. BBD System Performance

Conclusion

The use of an intraframe hybrid transformation composed of a horizontal unitary discrete cosine transform and a vertical first-order DPCM has been shown through simulation to have performance closely approximating that of a two-dimensional Karhunen-Loeve transform. This hybrid transform has been computed in real time with minimum complexity and memory through the use of LSI bucket brigade or charge coupled devices and conventional digital DPCM implementation.

Acknowledgment

This work was supported in part by Defense Advanced Research Projects Agency, Order Number 2303, Code Number 3G10, monitored by Col. H. M. Federhen.

References

1. Habibi, A. and Wintz, P. A., "Image coding by linear transformation and block quantization," IEEE Trans Commun. Technol., vol. COM-19, pp. 50-62, Feb. 1971.
2. Tasto, M. and Wintz, P. A., "A bound on the rate-distortion function and application to images," IEEE Trans. Inform. Theory, vol. IT-18, pp. 150-159, Jan. 1972.
3. Pearl, J., "Basis-Restricted Transformation and Performance Measures for Spectral Representations," IEEE Trans. on Inform. Theory, vol. IT-17, (6): pp. 751-752, Nov. 1971.
4. Davenport, W. B., and Root, W. L., Random Signals and Noise, McGraw-Hill: New York 1958, pp. 96-101.
5. Ray, W. D., and Driver, R. M., "Further decomposition of the Karhunen-Loeve series representation of a stationary random process," IEEE Trans. Inform. Theory, Vol. IT-16, pp. 663-668, Nov. 1970.
6. Pearl, J., "On coding and filtering stationary signals by discrete Fourier transforms," IEEE Trans. Inform. Theory, vol. IT-19, pp. 229-232, March 1973.
7. Ahmed, N., Natarajan, T., and Rao, K. R., "On image processing and a discrete cosine transform," IEEE Trans on Computers, C-23, pp. 90-03, Jan. 1974.
8. Habibi, A., "Hybrid Coding of Pictorial Data," Comm. Tech., COM-22, pp. 614-624, May 1974.
9. Shea, T. E., Transmission Networks and Wave Filters, D. Van Nostrand Co., 1929.
10. Bluestein, L. T., "A Linear Filtering Approach to the Computation of Discrete Fourier Transform," IEEE Trans. Audio and Electroacoustics, AU 18, pp. 451-455, 1970.
11. Rabiner, L. R., et al., "The Chirp-Z Transform Algorithm," IEEE Trans Audio and Electroacoustics, vol. AU-17, pp. 86-92, 1969.
12. Cooley, J. W. and Tukey, John W., "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, Vol. 19, No. 90, pp. 297-301, 1965.
13. Whitehouse, H. J., Speiser, J. M., and Means, R. W., "High Speed Serial Access Linear Transform Implementations," presented at the All Applications Digital Computer (AADC) Symposium, Orlando, Florida (January 23-25, 1973). Reprinted in Image Transmission via Spread Spectrum Techniques, ARPA Quarterly Technical Report, Order Number 2302 Code Number 3G 10 (February 1, 1973-June 1, 1973).
14. Buss, D. D., Collins, D. R., Bailey, W. H., and Reeves, C. R., "Transversal Filtering Using Charge Transfer Devices," IEEE J. of Solid State Circuits, SC-8, pp. 138-146, 1973.
15. Alsup, J. M., Means, R. W., and Whitehouse, H. J., "Real Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices," Proc. IEEE International Specialist Seminar on Component Performance and Systems Application of Surface Acoustic Wave Devices, Aviemore, Scotland (September 24-28, 1973).
16. Means, R. W., Buss, D. D., and Whitehouse, H. J., Real-Time Discrete Fourier Transforms Using Charge Transfer Devices, Proceedings of the CCD Applications Conference, Naval Electronics Laboratory Center, San Diego, Calif. Sept. 1973, pp. 95-101.

THE TECHNOLOGY OF CHARGE-COUPLED DEVICES
FOR VIDEO BANDWIDTH REDUCTION

D. D. Buss, R. W. Brodersen, C.R. Hewes
and A. F. Tasch, Jr.
Texas Instruments, Incorporated
Dallas, Texas 75222

Abstract

Charge-coupled devices (CCD's) are analog sampled-data delay lines which can be used to implement many of the filtering functions required for transform encoding in video bandwidth reduction. The CCD transversal filter is a fundamental building block which is particularly cost effective in terms of its simplicity and versatility. In particular it can be used to perform the Fourier transform of video signals by means of the chirp z-transform (CZT) algorithm. CCD's have performance limitations compared to digital filters, but for applications which fall within their performance ranges, CCD's offer advantages of lower cost, smaller size, lighter weight, lower power and improved reliability over digital filters.

I. Introduction

Charge-coupled devices (CCD's) are ideally suited to the implementation of many of the sampled-data filtering functions which are required for transform encoding of video images for bandwidth reduction. An example of the application of CCD's to video bandwidth reduction will be described in a subsequent paper¹. It is the purpose of this paper to review CCD filtering technology as it applies to this important problem.

CCD's have performance limitations compared to digital filters, but for applications which fall within their performance ranges, CCD's offer advantages of lower cost, smaller size, lighter weight, lower power and improved reliability over digital filters. In a sense CCD filters combine the best features of analog and digital implementations: time delays are precisely controlled by a master clock, but digitizing is eliminated.

The CCD transversal filter is a fundamental building block which is particularly cost effective in terms of its simplicity and versatility. In Section II the design, operation, and performance limitations of CCD transversal filters will be described and some examples will be given.

CCD's can be combined with other MOS circuitry to achieve complex signal processing functions. By combining CCD transversal filters with MOS multipliers, the chirp z-transform (CZT) algorithm for discrete Fourier transform (DFT) can be realized. The CCD CZT is an important function for video bandwidth reduction and will be described in Section III.

II. CCD Transversal Filters

The block diagram of a CCD transversal filter is given in Figure 1. It consists of M delay stages, D, together with circuitry for performing the weighted summation of node voltages v_k . Each delay stage consists of p transfer electrodes in a p-phase CCD. The filter is sampled and clocked at clock frequency f_c , and the z-transform characteristic $H(z)$ is given by

$$H(z) = \sum_{k=1}^M h_k z^{-k} \quad (1)$$

where h_k , $k=1, M$ are the weighting coefficients and z is related to frequency f through the relation

$$z = \exp [i2\pi f/f_c] \quad (2)$$

Filter Design

The CCD transversal filter is an important component primarily because the weighting and summing circuitry is quite simple. Using the split electrode weighting technique,² a CCD delay line can be made into a transversal filter by splitting one of the electrodes in each delay stage and putting a differential current integrator (DCI) in the clock line as shown schematically in Figure 2.

In Figure 2 the signal charges are sampled on the ϕ_3 clock electrodes. As charge transfers from the k^{th} ϕ_2 electrode to the k^{th} ϕ_3 electrode, the current which flows to the k^{th} ϕ_3 electrode line consists of a part which would flow if no signal charge were present plus a part approximately equal to the signal charge. Therefore the signal charge can be determined by integrating the current flowing to the ϕ_3 electrode during charge transfer. Weighting is performed by splitting the ϕ_3 electrode and integrating separately the current flowing to each portion. A weighting coefficient of zero corresponds to a split in the center of the electrode, and positive and negative h_k are achieved by appropriately proportioning the charge between $\phi_3^{(+)}$ and $\phi_3^{(-)}$. The summation is achieved by connecting together the $\phi_3^{(+)}$ electrode and the $\phi_3^{(-)}$ electrodes as shown in Figure 2(b) and the filter output is obtained by integrating and differencing the $\phi_3^{(+)}$ and $\phi_3^{(-)}$ clock line currents in the output DCI as shown.

The split electrode technique was first developed for use with bucket-brigade devices (BBD's)³ and has been widely used for both CCD and BBD filters.^{4,5,6,7,8,9} The design techniques are further described in References 2 and 9.

Limitations

CCD transversal filters have a number of important performance limitations, each of which will be discussed.

Maximum clock rate. In spite of the fact that CCD's have been operated at clock frequencies in excess of 100 MHz,¹⁰ practical CCD transversal filters are limited to 20 MHz or less. This limitation results from the non-CCD electronics (clock drivers, summing and weighting circuitry etc.) but is no less of a limitation because of that.

Minimum clock rate. The slowest that a CCD can be clocked depends upon thermal leakage. An M-stage CCD clocked at frequency f_c is required to hold charge for a total delay time $T_d = M/f_c$. The CCD gradually loses its ability to hold signals, however, due to thermal leakage, and the potential wells are completely filled within the storage time t_s which is typically $t_s \approx 50$ sec. at room temperature. The minimum clock frequency is therefore determined by the requirement that $T_d \ll t_s$. Delay times of up to $T_d \approx 1$ sec. are feasible at room temperature. However, leakage current increases by a factor of two for each 8°C increase in temperature, and the minimum clock frequency increases correspondingly.

Charge transfer loss. Transfer of charge from one electrode to the next is not perfect and each time charge transfers a fraction α is left behind. For a p-phase CCD, the loss per delay stage is

$$\epsilon = p\alpha \quad (3)$$

and the z-transform characteristic of each delay stage becomes

$$H_D(z) = \left(\frac{1 - \epsilon}{1 - \epsilon z^{-1}} \right) z^{-1} \quad (4)$$

$$\approx z^{-1} \exp[-\epsilon(1 - z^{-1})] \quad (5)$$

A filter characteristic such as that given in equation (1) is therefore modified by charge transfer loss to

$$H(z) = \sum_{k=1}^m h_k \left(\frac{1 - \epsilon}{1 - \epsilon z^{-1}} \right)^k z^{-k} \quad (6)$$

One way¹¹ of viewing this result of non-zero ϵ is to take the ideal transfer function $H^1(f)$ and replace f by

$$f' = f + \frac{\epsilon f_c}{2\pi} \left\{ \sin(2\pi f/f_c) - i[1 - \cos(2\pi f/f_c)] \right\} \quad (7)$$

A scaling of the frequency axis results from the real part of f' , and an attenuation of the frequency response results from the imaginary part.

Another way¹² of viewing the effect of non-zero ϵ is to move all the zeros z_k (and poles of the filter is recursive) of the filter to new position given by

$$z_k^1 = \epsilon + (1 - \epsilon) z_k \quad (8)$$

For example, a bandpass filter having passband at $f_o = f_c/4$ has zeros on the unit circle as shown in Figure 3. The effect of non-zero ϵ is to shift the circle to the right by ϵ and decrease its radius by ϵ . Geometric analysis of Figure 3 indicates that the principle features of the bandpass frequency response can be approximated by

$$H_{\epsilon}(f) \approx H^1(f + \Delta f) A(f) \quad (9)$$

where the frequency shift Δf is given by

$$\Delta f = \frac{\epsilon f_c}{2\pi} \sin(2\pi f/f_c) \quad (10)$$

and the envelope function $A(f)$ which attenuates the local maxima is given to first order in $M\epsilon$ by

$$A(f) = 1 - \frac{M}{2} [1 - \cos(2\pi f/f_c)] \quad (11)$$

The loss parameter ϵ is typically on the order of $\epsilon \approx 10^{-4}$ and typical filtering applications require $M\epsilon \leq .1$. Thus, filters having up to $M = 10^3$ stages are possible without resorting to the complication of multiplexing.

Linearity. Design techniques exist for inputting electronic charge q_n into a CCD with good linearity between q_n and input voltage.¹³ However, the split electrode output circuitry does not sense q_n but rather $q_s = q_n + q_b$ where q_b is the depletion layer bulk charge. Therefore, for filters using the split electrode output, the input should be designed so that the total charge q_s is proportional to the input signal. A further discussion of filter linearity is found in Reference (9).

Noise. Noise in CCD transversal filters has two sources (1) noise within the CCD itself and (2) noise associated with the DCI. The CCD is inherently a low noise device¹⁴ especially if buried channel CCD filters are used. In addition, the weighted summation performed within the filter gives processing gain against CCD noise in much the same way as it does against input noise. On the other hand, DCI noise is added after the weighted summation, and for this reason it usually dominates filter noise.

The principal sources of DCI noise are (1) kTC noise¹⁴ on the split electrode clock lines and (2) voltage amplifier noise. Filters tested to date have been limited by voltage amplifier noise. However, by improving the amplifiers, 80 dB dynamic range should be achievable.

Weighting Coefficient Error. Error in the weighting coefficients is an important limitation of CCD transversal filters for spectral analysis applications because it limits the out-of-band rejection or sidelobe suppression which can be achieved. Errors result from; stepping quantization error in photomask generation, photomask misalignment, non-uniform etching of the split electrodes, non-uniform oxide thickness, and non-uniform substrate resistivity. All of these errors can be reduced by making wide CCD channels at the expense of larger size and higher clock line capacitance. Filters made to date with 5 mil. wide channels have weighting coefficient error on the order of .5%.⁹

If the weighting coefficients h_k , $k = 1, M$ each have error ξ_k , $k = 1, M$. The ideal frequency characteristic $H^1(f)$ has added to it an error $\Delta H(f)$ given by

$$\Delta H(f) = \sum_{k=1}^M \xi_k e^{-i2\pi k f/f_c} \quad (12)$$

Since the ξ_k are not known, it is convenient to treat them as random variables. The assumption of statistical independence applies to stepping quantization error, and gives the result for bandpass filters¹⁵

$$\frac{\Delta H_{rms}}{H^1(f_o)} \approx \frac{2\xi_{rms}}{\sqrt{M} \bar{w}} \quad (13)$$

where \bar{w} is the average value of window function used to weight the filter. As an example, consider a 500-stage bandpass filter which is characterized by $M = 500$, $\bar{w} = .5$, and $\xi_{rms} = .005$. Eq (13) predicts that the rms frequency response error due to weighting coefficient error is 60 dB below the amplitude of the response in the passband. Weighting coefficient error is considered in further detail in References (9) and (16).

Examples

The frequency response of a 500-stage CCD bandpass filter is shown in Figure 4. The weighting coefficients are of the form

$$h_k = w_k \sin\left(\frac{\pi k}{2}\right) \quad k = 1, 500 \quad (14)$$

where

$$w_k = 0.54 + .46 \cos \left[2\pi(k - 250)/500 \right] \quad (15)$$

the 3 dB bandwidth is

$$\frac{\Delta f}{f_0} = .011 \quad (16)$$

and the highest sidelobes are -41 dB compared to -42.8 dB for the ideal case. The characteristics of this filter are described further in Reference (9).

Figure 5 shows the impulse response and correlation response of a 100-stage CCD filter matched to a pseudorandom pn sequence code. The highest time sidelobe is 16 dB below the correlation peak.

III. CCD Chirp z-Transform

The chirp z-transform (CZT) is an algorithm for performing the discrete fourier transform (DFT) in which the bulk of the computation is performed in a chirp transversal filter, and for this reason it is particularly attractive for CCO implementation.^{17,18}

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi nk/N} \quad (17)$$

and using the substitution

$$2nk = n^2 + k^2 - (n - k)^2 \quad (18)$$

the following equation results:

$$X_k = e^{-i\pi k^2/N} \left[\sum_{n=0}^{N-1} \left(x_n e^{-i\pi n^2/N} \right) e^{i\pi(k-n)^2/N} \right] \quad (19)$$

This equation has been factored to emphasize the three operations which make up the CZT algorithm; premultiplication by a chirp, filtering in a chirp transversal filter and postmultiplication by a chirp. When only the power density spectrum is required, the postmultiplication by $e^{-i\pi k^2/N}$ can be eliminated, and a block diagram of the circuit implementation is given in Figure 6.

The 200-stage BBO filters shown in Figure 7 were the first charge transfer devices used to demonstrate the CZT concept.¹⁹ The weighting coefficients are samples of the chirp or linear FM functions

$$\begin{aligned} h(t) &= \cos \mu t^2 \\ g(t) &= \sin \mu t^2 \end{aligned} \quad -\frac{T_d}{2} < t < \frac{T_d}{2} \quad (20)$$

The charge transfer loss of BBD's is somewhat greater than that of CCO's, and the devices shown in Figure 7 have $\epsilon \approx 2 \times 10^{-3}$ which attenuates the impulse response visibly. In spite of the high MC product the devices performed adequately because the CZT output is not particularly sensitive to charge transfer loss.⁹

500-stage CCO filters have been used to implement the block diagram of Figure 6²⁰ with the result shown in Figure 8. The CCO CZT is discussed further in References (9) and (21).

IV. Conclusions

The importance of the CCD CZT to video transform encoding has been recognized,²² and an example of this application will be discussed in a subsequent paper¹. It has been the purpose of this paper is to review the current status of CCD technology as it applies to filtering for video bandwidth reduction.

CCD's have performance limitations which have been briefly discussed. It appears that transform encoding for many practical systems can be designed within these performance limitations and therefore, CCD's are expected to be important in this application. Because of its extreme simplicity, the CCD transversal filter and the CCD CZT are expected to be key building blocks in transform encoding systems of the future.

Acknowledgement

This paper has reviewed CCD development performed under the following government contracts.

Naval Electronics Command, Contract N00039-73-C-0013 monitored by Or. David F. Barbe.

Rome Air Development Center, Contract F30602-73-C-0027 monitored by Charles N. Meyer.

U.S. Army Electronics Command, Contract DAAB07-73-C-0351, monitored by Robert H. Sproat.

U.S. Army Electronics Command, Contract DAAB07-73-C-0266, monitored by Ted J. Lukaszek.

Naval Undersea Center, Contract N00123-74-C-1366, monitored by Dr. Robert W. Means.

The authors express their sincere appreciation to the many people at Texas Instruments whose efforts have made this paper possible.

References

1. "Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices," H.J. Whitehouse, R. W. Means, and E. H. Wrench, This Conference.
2. D.D. Buss, D.R. Collins, W.H. Bailey, and C.R. Reeves, "Transversal Filtering Using Charge Transfer Devices," IEEE J. Solid-State Circuits, SC-8, pp 134-146, April 1973.
3. F.L.J. Sangster, "The Bucket-Brigade Delay Line, A Shift Register for Analogue Signals," Philips Technical Review, 31, pp 97-110, 1970.
4. R.D. Baertsch, W.E. Engeler, H.S. Goldberg, C.M. Puckette, and J.J. Tiemann, "Two Classes of Charge Transfer Devices for Signal Processing," Proc. International Conference Technology and Application of CCD's, Edinburgh, Sept. 1974, pp 229-236.
5. D.D. Buss and W.H. Bailey, "Application of Charge Transfer Devices to Communication," Proc. CCD Applications Conference, San Diego, Sept. 1973, pp 83-93.
6. A. Ibrahim and L. Sellars, "CCD's for Transversal Filters Applications," IEEE International Electron Devices Meeting, Technical Digest, Washington, Dec. 1974, pp, 240-243.
7. J.A. Sekula, P.R. Prince and C.S. Wang, "Non-Recursive Matched Filters Using Charge-Coupled Devices" IEEE International Electron Devices Meeting, Technical Digest, Washington, Dec. 1974, pp 244-247.
8. M.F. Tompsett, A.M. Mohsen, D.A. Sealer, and C.H. Séquin, "Design and Characterization of CCD's for Analog Signal Processing," Same as Reference 8, pp 83-89.
9. R.W. Brodersen, C.R. Hewes and D.D. Buss, "Spectral Filtering and Fourier A Analyses Using CCD's," IEEE Advanced Solid-State Components for Signal Processing, IEEE International Symposium on Circuits and Systems, Newton, Mass, April 1975, pp 43-68.

10. L.J.M. Esser, M.G. Collet, and J.G. Van-Santen, "The Peristaltic Charge Coupled Device," 1973 International Electron Devices Meeting, Technical Digest, pp 17-20.
11. D.D. Buss, W.H. Bailey, and D.R. Collins, "Analysis and Applications of Analog CCD Circuits," Proc. of International Symp. on Circuit Theory, Toronto, pp 3-7, April 1973.
12. A. Gersho and B. Gopinath, "Filtering with Charge Transfer Devices," Proc. 1975 IEEE International Symposium on Circuits and Systems, Newton, Mass., April 1975, pp 183-186.
13. C.H. Sequin and A.M. Mohsen, "Linearity of Electrical Charge Injection into Charge Coupled Devices," IEDM Technical Digest, Washington, D.C., p 229, 1974.
14. J.E. Carnes and W.F. Kosonocky, "Noise Sources in Charge Coupled Devices," RCA Review 33, p 327, June 1972.
15. D.D. Buss, C.R. Reeves, W.H. Bailey, and D.R. Collins, "Charge Transfer Devices in Frequency Filtering," Proc. of 26th Annual Symposium on Frequency Control, Atlantic City, pp 171-179, June 1972.
16. C.M. Puckette, W.J. Butler, and D.A. Smith, "Bucket-Brigade Transversal Filters," IEEE Trans. on Comm. COM-22, pp 926-934, July 1974.
17. L.R. Rabiner, R.W. Schafer, and C.M. Rader, "The Chirp z-Transform Algorithm," IEEE Trans. on Audio and Electroacoustics, AU-17, pp 86-92, June 1969.
18. H.J. Whitehouse, J.M. Spelser and R.W. Means, "High Speed Serial Access Linear Transform Implementations," presented at the All Applications Digital Computer Symposium, Orlando, Florida, Jan. 1973. NUC TN 1026.
19. "Real Time Discrete Fourier Transforms Using Charge Transfer Devices," R.W. Means, D.D. Buss and H.J. Whitehouse, Proc. CCD Applications Conference, San Diego, Sept. 1973, pp 95-101.
20. R.W. Brodersen, H.S. Fu, R.C. Frye and D.D. Buss, "A 500-Point Fourier Transform using Charge-Coupled Devices," 1975 IEEE International Solid-State Circuits Conference Digest of Technical Papers, Philadelphia, Feb. 1975, pp 144-145.
21. H.J. Whitehouse, R.W. Means, and J.M. Speiser, "Signal Processing Architectures Using Transversal Filter Technology," IEEE Advanced Solid-State Components for Signal Processing, IEEE International Symposium on Circuits and Systems, Newton, Mass., April 1975, pp 5-29.
22. R.W. Means, H.J. Whitehouse and J.M. Speiser, "Television Encoding Using a Hybrid Discrete Cosine Transform and A Differential Pulse Code Modulator in Real Time," 1974 National Telecommunications Conference Record, San Diego, Dec. 1974, pp. 69-74.

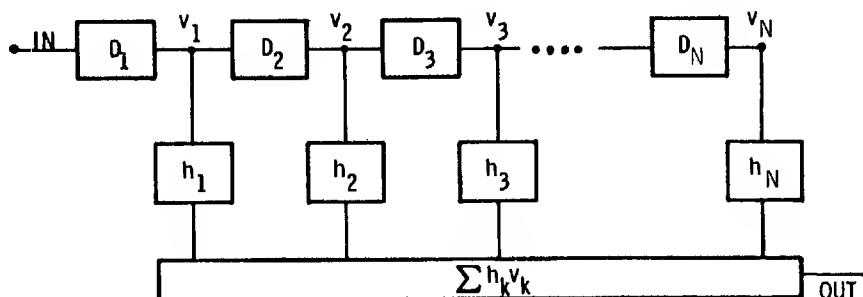


Figure 1. Block diagram of a transversal filter consisting of M delay stages D_k and M weighting coefficients h_k .

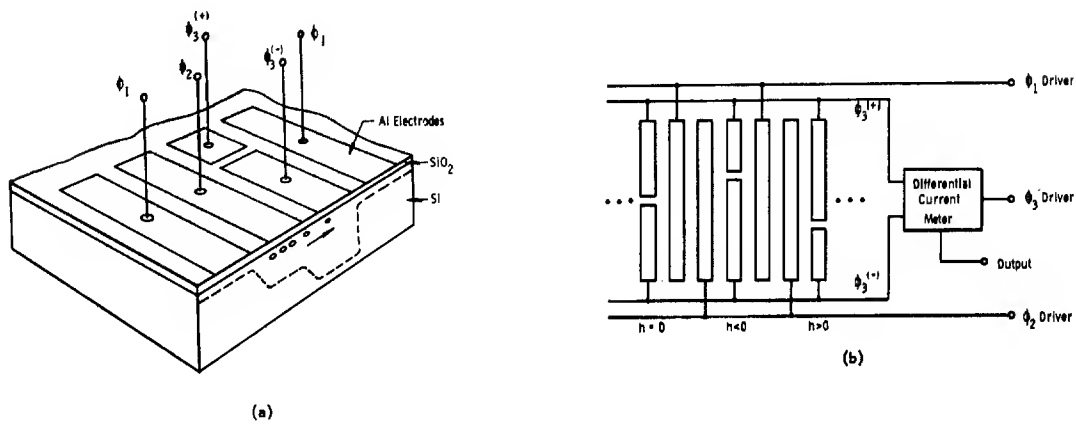


Figure 2. Schematic of the split electrode technique for weighting coefficients of transversal filter.

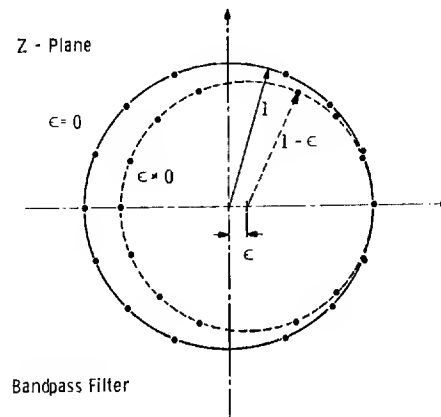


Figure 3. Zeros of the transfer function of a bandpass filter in the z-plane. Zeros are on the unit circle (solid) with no charge transfer loss, $\epsilon = 0$, with charge transfer loss $\epsilon > 0$ the zeros move to positions on the contracted and shifted circle shown with a dashed curve.

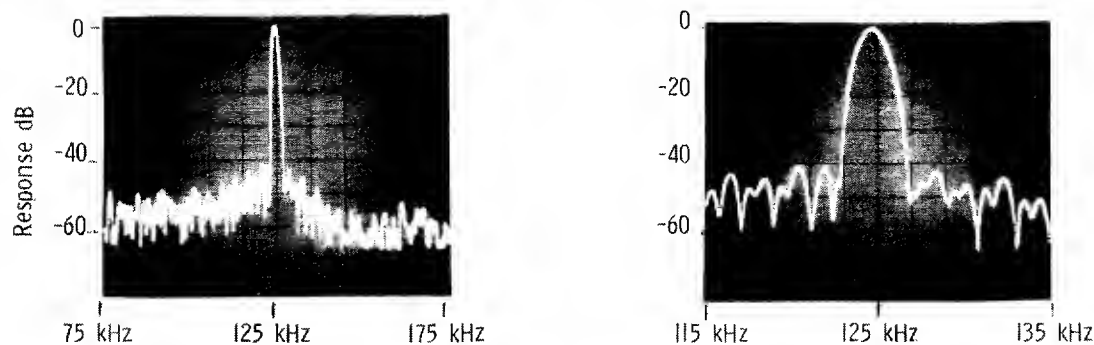


Figure 4. Measured frequency response of a 500-stage CCD bandpass filter having the Hamming window function.

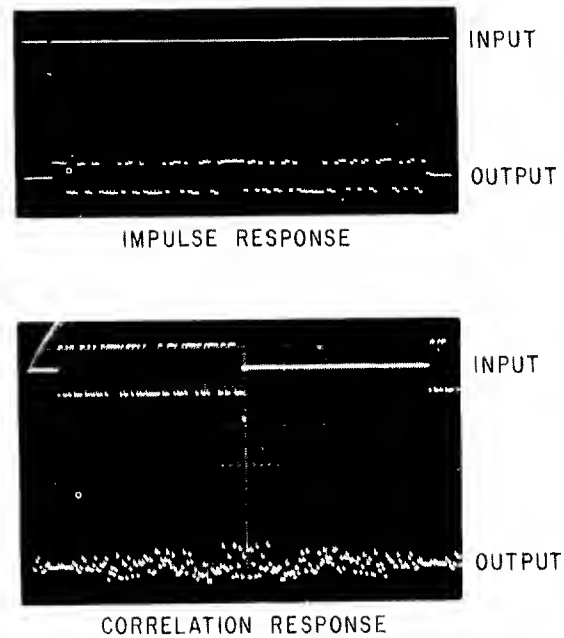


Figure 5. Impulse response (top) and correlation response (bottom) of a 100-stage, pn sequence CCD filter. Note the correlation peak in the output waveform.

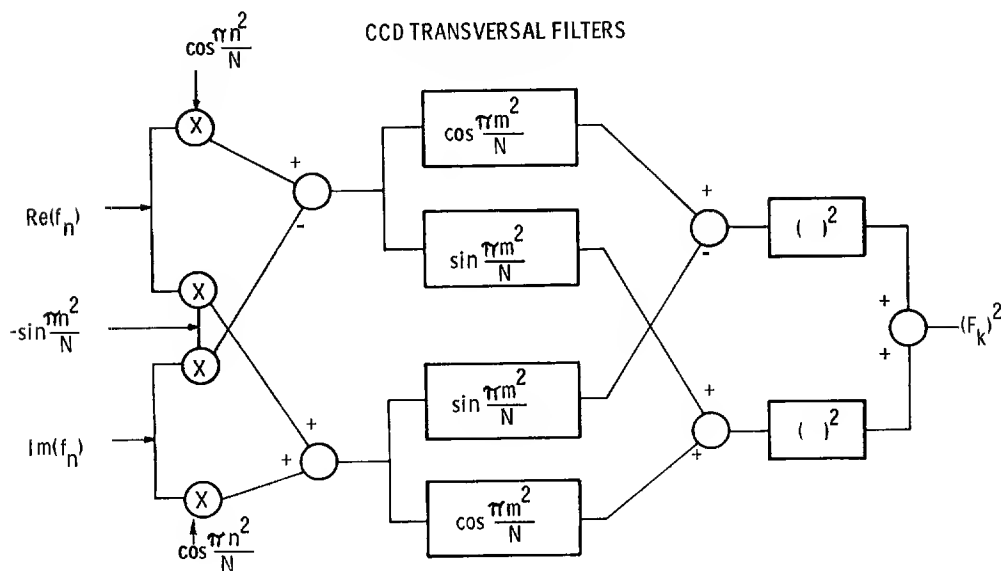


Figure 6. Block diagram of the CCD CZT.

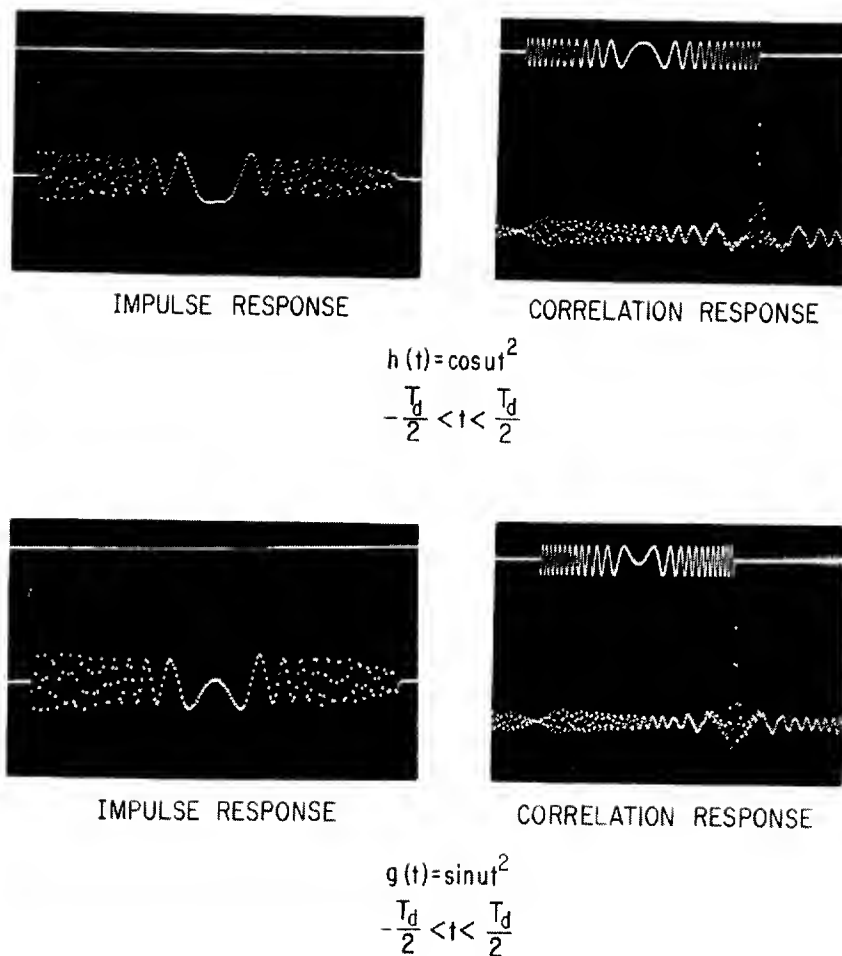


Figure 7. Impulse responses and correlation responses of two, 200-stage BBD filters used to implement the CZT (See Reference (19)).

Input: Frequency sweeps
 50 kHz to 100 kHz in
 $T_d = .11$ msec.
 CZT Output: Line spectrum
 having period
 $T_d^{-1} = 9$ kHz

Square root of the
 power density spectrum
 displayed using an HP
 spectrum analyzer

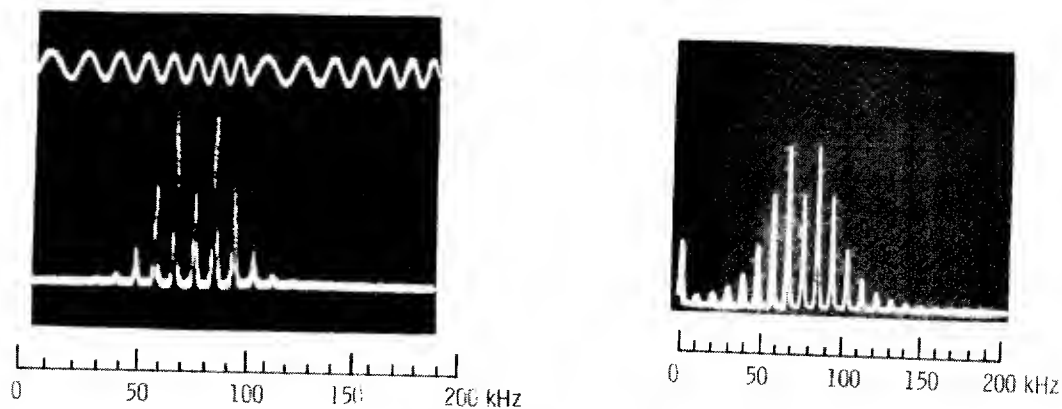


Figure 8. Performance of system of Figure 6 using 500-stage CCD filters. (See Reference (20)).

Session II
ADVANCED IMPLEMENTATIONS

Chairman
ALI HABIBI
TRW Systems Group

REAL-TIME VIDEO COMPRESSION ALGORITHM FOR HADAMARD TRANSFORM PROCESSING

Scott C. Knauer
Ames Research Center, NASA
Moffett Field, California 94035

Abstract

A real-time digital video processor using Hadamard transform techniques to reduce video bandwidth is described. The processor can be programmed with different parameters to investigate various algorithms for bandwidth compression. The processor is also adaptive in that it can select different parameter sets to trade off spatial resolution for temporal resolution in the regions of the picture that are moving. Algorithms used in programming the system are described along with results achieved at various levels of compression. The algorithms relate to spatial compression, temporal compression, and the adaptive selection of parameter sets.

Introduction

To evaluate the effectiveness of a noninformation preserving video compression scheme, a measure of picture quality must be made at each level of compression. In the case of television intended for a human audience, the eye is the final judge of quality. The real-time Hadamard system at Ames¹ was developed to allow algorithms to be tested with a large amount of material (any NTSC video source) and with immediate visual feedback for the experimenter. The effects of various transform algorithms on motion may be observed for indefinitely long or complex sequences of movement, and adaptive algorithms, trading off spatial resolution against temporal resolution, are easily implemented and evaluated.

Basic SystemPicture Representation

Incoming video is sampled at 8 megasamples/sec or 512 samples, called picture elements or pels, per line. Each pel's brightness is represented by a 6-bit two's complement binary number with black negative, white positive, and grey at zero. Four frames (8 fields) of the digitized video are stored, and the resulting 512x525x4 matrix of 6-bit numbers, shown in Fig. 1(a), is then partitioned into a set of subpictures as shown in Fig. 1(b). All picture processing and compression is done within these 4x4x4-pel subpictures.

Hadamard Transform

The Hadamard transform of the spatial information in the 4x4x4 subpicture can be considered by looking at a slice of the original subpicture that is 4 pels horizontally by 4 pels vertically. This 4x4 subpicture can be represented by 16 basis vectors, each describing the position of a pel in the 4x4 subpicture. The vector coefficients represent the brightness of their respective pels. This is the basic format of the picture information in digitized television.

A set of Hadamard basis vectors, shown in Fig. 2(a), can also be used to describe the 4x4 subpicture. The subscripts of the basis vectors \bar{A}_{vh} indicate the number of black-to-white or white-to-black transitions in the vertical (v) and horizontal (h) directions, respectively. (The significance of the Q's will be explained later.) The coefficients of each Hadamard vector measure the degree to which the pattern shown in Fig. 2(a) is present in the 4x4 subpicture. Each coefficient may be calculated by comparing the incoming 4x4 to the vector pattern in Fig. 2(a) and noting that:

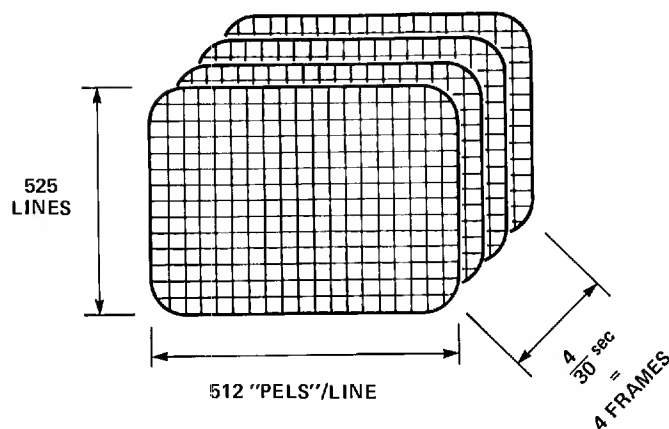
$$\frac{1}{16} \left\{ \left(\begin{array}{l} \text{Sum of pel amplitudes in} \\ \text{positions corresponding} \\ \text{to black areas of } \bar{A}_{vh} \end{array} \right) - \left(\begin{array}{l} \text{Sum of pel amplitudes in} \\ \text{positions corresponding} \\ \text{to white areas of } \bar{A}_{vh} \end{array} \right) \right\} = \text{Coefficient of } \bar{A}_{vh} \quad (1)$$

For the positive or negative magnitude of the coefficient of a given \bar{A}_{vh} to be large, the subpicture must have somewhat the same pattern of contrast as \bar{A}_{vh} or its negative.

The extension of this to the full 4x4x4 subpicture is straightforward. The subscripts of the 64 vectors \bar{V}_{tvh} indicate the number of transitions in the temporal (t), vertical (v), and horizontal (h) dimensions. The number of transitions in a given direction is the horizontal, vertical, or temporal sequence of a vector. Figure 2(b) illustrates the patterns for the 16 vectors with no vertical transitions, \bar{V}_{t0h} , as well as a few of the more significant vectors with vertical transitions. The latter show that the pattern of transitions is the same in each dimension. Equation (1) remains valid with 1/64 replacing 1/16 and \bar{V}_{tvh} replacing \bar{A}_{vh} .

More detailed and precise treatments of the application of Hadamard transforms to picture processing are given by Harmuth,² Pratt, Kane, and Andrews,³ and others.

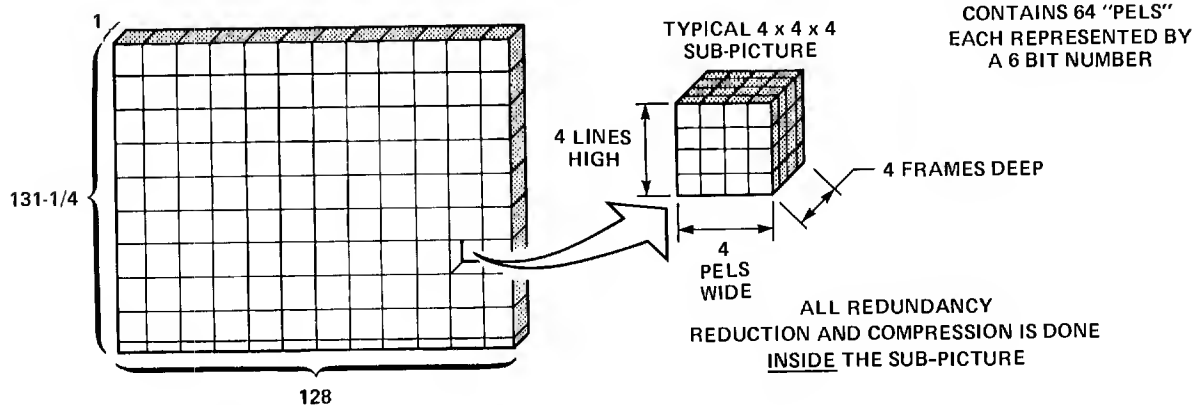
DIGITIZE PICTURE AND SAVE 4 FRAMES



WE HAVE 4 512 x 525 MATRICES – OR
ONE 512 x 525 x 4 MATRIX – REPRESENTING
4 STILL PICTURES OR
4/30 sec OF REAL TIME TV

THE BRIGHTNESS OF EACH PICTURE
ELEMENT OR "PEL" IS REPRESENTED
BY A 6 BIT BINARY NUMBER

DIVIDE "MATRIX" INTO 4 x 4 x 4 SUB-MATRICES OR SUB-PICTURES



ALL REDUNDANCY
REDUCTION AND COMPRESSION IS DONE
INSIDE THE SUB-PICTURE

Fig. 1. Digitization and Storage of Video Data prior to transform.

Q = NO. OF QUANTIZATION BITS
ASSOCIATED WITH EACH
BASIS VECTOR

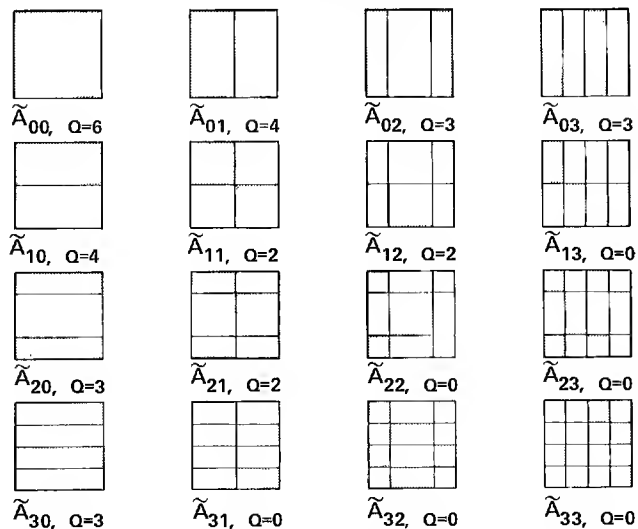


Fig. 2(a). Two-dimensional Hadamard basis vector representation of a 4x4 pel subpicture.

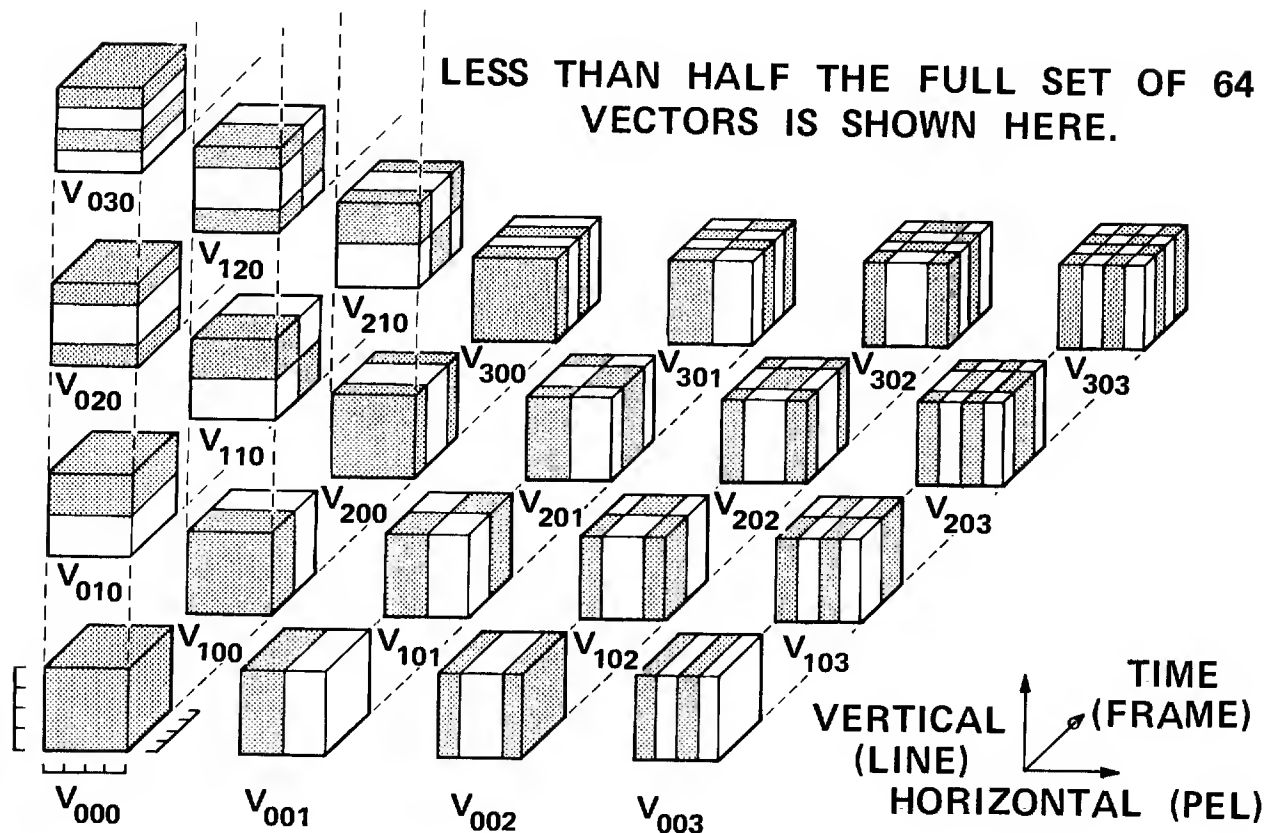


Fig. 2(b). Three-dimensional Hadamard basis vector representation of a $4 \times 4 \times 4$ pel subpicture.

The video processor takes successive 64-sample subpictures (left to right and top to bottom in Fig. 1(b)) and performs a 64-point parallel Hadamard transform on each one in turn, yielding sets of 64 Hadamard vectors. Six successive stages of addition or subtraction are necessary to do this, requiring that the binary numbers grow to 12 bits. However, after two operations, the numbers are kept at an 8-bit length by truncating the least significant bit in the next four operations.

Quantization

Sixty-four 8-bit numbers are no improvement over 64 6-bit numbers. However, the 64 Hadamard coefficients, unlike the 64 individual pel intensities, are not all equally important. Over half are quite close to zero in amplitude and can be replaced by zero; large numbers of tiny flashing checkerboards do not appear in common TV pictures. Some other vectors, too large to throw away, still do not warrant the use of 8-bits to represent them. Figure 3 gives an example of an 8-bit number quantized to 3 bits. The initial range of 2^8 or 256 possible amplitude values is divided into 8 segments by 7 cutpoints symmetric about zero. The incoming vector values in each segment are replaced by the representative value assigned to that segment. Thus, a 3-bit code can represent the 8 segments and, after transmission, can be used to look up the proper representative value. The cost of this bit reduction is the quantization error introduced in the vector coefficient.

The processor has the capability to quantize 31 different vectors; V_{000} is retained with 7 bits (one bit truncated), and the remainder are set to zero. The 31 vectors may be quantized to 4, 3, or 2 bits by the experimenter. The experimenter enters, on thumbwheel switches, the three subscripts of the vector to be quantized, its cutpoints, and its representative values. When the "program" button is pushed, the values of the switches are stored in a solid-state memory in the processor, and the next vector may be entered. Four of the 31 vectors can be programmed to escape quantization and be represented by 7 bits, the least significant bit being truncated.

All together up to four different sets or "options" of 31 vector programs each may be stored. An option switch is set at the time each vector is programmed. During picture processing, the option switch allows the experimenter to compare the fidelity of the picture as quantized by the different vector sets stored in each option. Additional hardware allows the machine to select different options itself, or to "adapt" in response to the amount of motion sensed in each subpicture. For this application the different options are programmed to trade off spatial fidelity for temporal fidelity as the degree of motion increases.

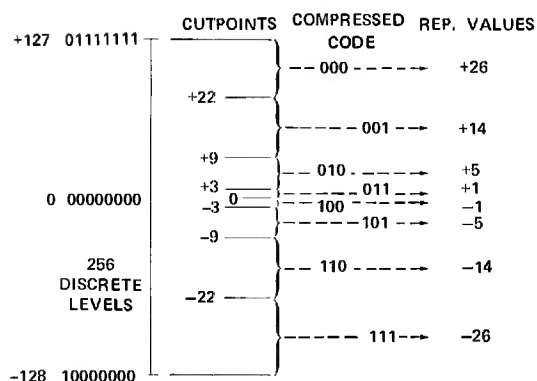


Fig. 3. Compression: how to reduce an 8-bit number.

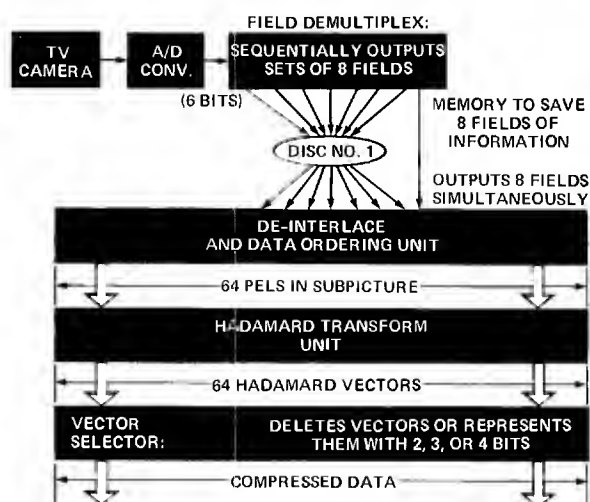


Fig. 4(a). Three-dimensional Hadamard video compression encoder subsystem.

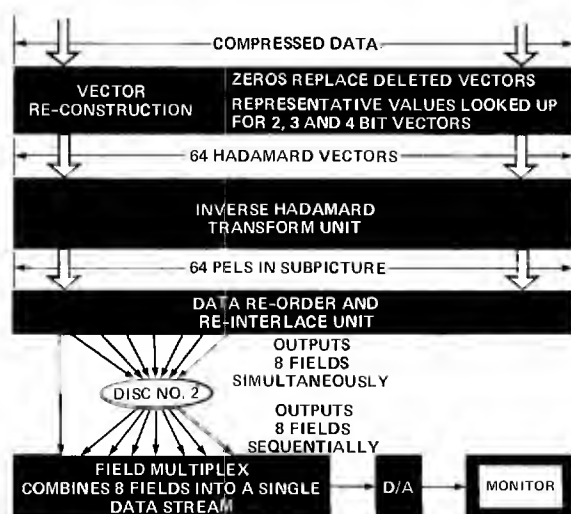


Fig. 4(b). Three-dimensional Hadamard video compression decoder subsystem.

After the vectors have been represented by 7, 4, 3, or 2 bits, or set to zero, they are processed by the decode Hadamard transformer (the Hadamard transform is self-inverse). The resulting pel amplitudes are rearranged into proper television format and fed into a digital-to-analog converter and finally to a monitor.

Hardware

A block diagram of the hardware is shown in Figs. 4(a) and (b). Analog video is fed to the analog-to-digital converter. The field demultiplexer routes the first 7 fields of video to 7 sets of 6 tracks each (1 track/bit) on the disk. When the 8th field is routed to the de-interlace unit, the other 7 fields are read out of the disk simultaneously. The output of this unit consists of bursts of 64 pels, the subpicture format; data are then transformed. Quantization and the programming of the quantizer algorithms are done in the vector select and reconstruction blocks. The rest of the decode operation is essentially a mirror image of the encode system.

This brute force parallel implementation was chosen to simplify timing and thus maximize the flexibility of the system as a laboratory instrument. Serial designs are being worked out for specific vector sets and adaptive algorithms that were checked out using the laboratory processor. These require memories smaller than 2 fields and far less arithmetic hardware to build the encode or decode unit.

Algorithms

Algorithms for the Ames video processor consist of the sets of vector programs - cutpoints and representative values - that are used to code the picture information. The decisions that must be made in selecting a vector set are: (1) the vectors that are to be discarded, that is, set to zero; (2) the number of bits that will represent each one of the vectors kept; and (3) the location of the cutpoints and representative values for each vector.

The total number of bits allowed for representing a given 4x4x4 subpicture is set by the compression ratio or number of bits per pel desired. Within this constraint, the vector set is selected to minimize both temporal and spatial picture distortion, although more emphasis may be placed on the temporal or spatial fidelity, depending on the nature of the pictures and the requirements of the user.

One guideline in building vector sets is given by observing the probability distributions of the Hadamard vectors; the distributions indicate the energy in each pattern.

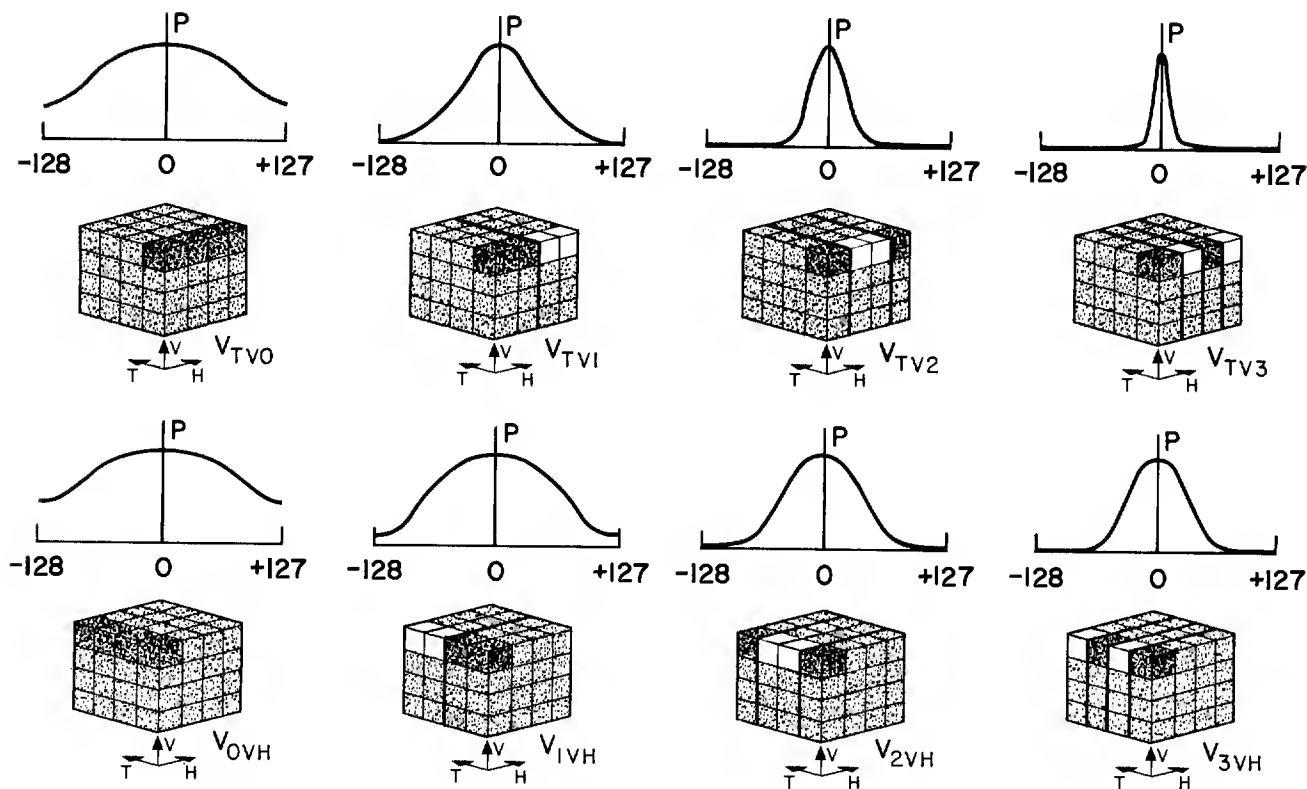


Fig. 5. Vector amplitudes as functions of horizontal and temporal sequence.

Probability Distribution of Vectors

Figure 5 shows the probability of occurrence of a vector coefficient as a function of its amplitude. The probability distributions of Hadamard vectors show that large vector amplitudes become rare as the vector's frequency increases in the horizontal or vertical direction in still pictures. This has been shown by picture measurements.⁴ Probabilities for the occurrence of large vector amplitudes also fall off with increasing temporal sequence, but not as rapidly (Fig. 5(b)). This is due to the nature of the source; very fine spatial detail (that would create larger values of high frequency spatial "checkerboard" vectors) is not as common as motion spanning about 20-50 subpictures in 4 frames (which creates larger values of high frequency temporal vectors).

Vertical-Temporal Interactions in Rapid Movement. When the picture is moving, large-amplitude probability still drops with increasing horizontal frequency. Probability does not vary as simply with vertical frequency. This is due to the fact that the two fields re-interlaced to form a frame are really 1/60 of a second apart in time, and vectors with a vertical frequency of 3, V_{t3h} , also contain temporal information (Fig. 6(a)). A high value of V_{030} , for instance, may mean the subpicture contains a pattern of horizontal stripes, or an abrupt motion in 1/60 of a second - not an unusual occurrence. These "vertical-temporal" interactions result from the necessity of dealing with the interlaced scan in NTSC video. Re-interlacing is necessary to exploit the close correlation of the information in adjacent fields.

For representing motion at very high speeds, it is useful to think of the subpictures as 4 pels \times 2 lines \times 8 fields, instead of 4 pels \times 4 lines \times 4 frames. The field-to-field temporal changes across 8 fields can be represented by 8 vectors. These vectors, given in the subscript notation of the 4 \times 4 \times 4 model are: V_{000} , V_{100} , V_{200} , V_{300} , V_{030} , V_{130} , V_{230} , and V_{330} .

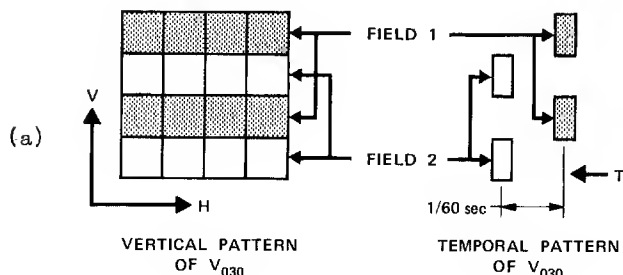


Fig. 6. Vertical-temporal vectors.

V_{330} has a similarity to V_{300} that is not apparent in the cubic pictorial representation, but can be seen in the representation of Fig. 6(b), in which the lines are shown in their proper positions in time and space. If frame 1 was formed from fields 2 and 3 instead of fields 1 and 2, and field 9 was dark, the V_{330} pattern would become a V_{300} pattern. V_{310} and V_{320} have a similar "phase shift" relationship.

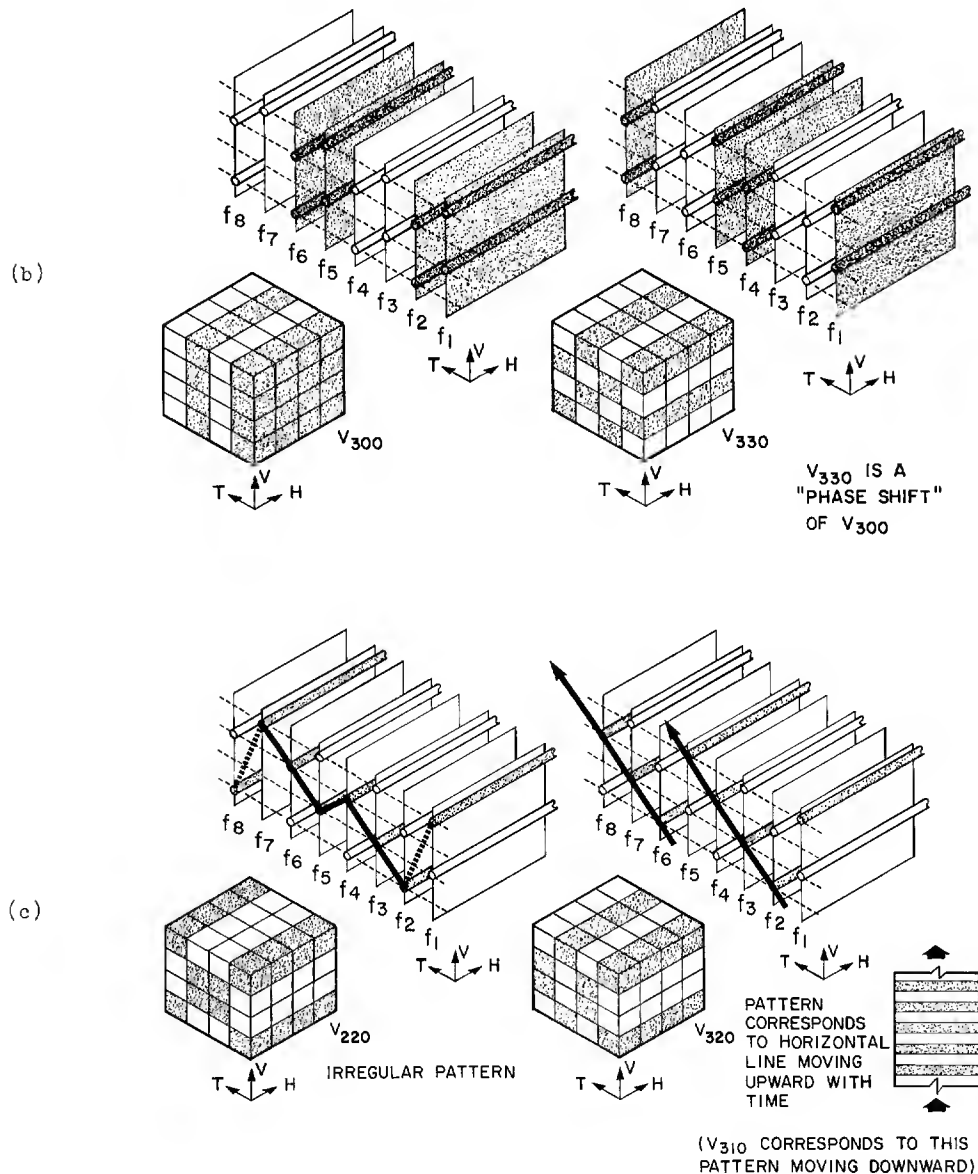


Fig. 6. Concluded.

A more subtle example of vertical-temporal effects is shown in Fig. 6(c). Experimental observation shows V_{220} to have fewer high amplitude occurrences than V_{320} ; Fig. 6(c) gives a possible explanation for this by showing that V_{320} corresponds to a more regular type of motion than V_{220} , when looked at with the lines staggered as they really are. Studies correlating various types of regular motion with vertical-temporal vectors are still underway.

Precedence Relation

The biggest problem in constructing vector sets is that of avoiding errors that introduce distracting visual noise. This is worse than having poor resolution, an expected result of lower bit rates. A picture represented by V_{000} - 7 bits and V_{100} , V_{001} , V_{010} - 4 bits each (about 0.28 bits per pel total) shows decreased resolution. Adding V_{200} - 4 bits (0.36 bits per pel total) does not improve the picture, but adds leading and trailing flashing "ghosts" to moving objects. Adding V_{300} - 4 bits solves the ghost problem, and the pair V_{200} - V_{300} improves the temporal resolution.

A precedence relation has been developed empirically to solve the problem of ghosts and noise. Vector sets built in accordance with this relationship will trade off resolution vs. bits per pel without generating distracting errors.

Figure 7(a) shows a precedence diagram for spatial vectors; arrows run from predecessors to successors. Any vector included in a set should have all its predecessors in the set. Vectors V_{002} - V_{003} and V_{020} - V_{030s} are pairs, and if one member of a pair is present, the other must be also. The "s" in V_{030s} refers

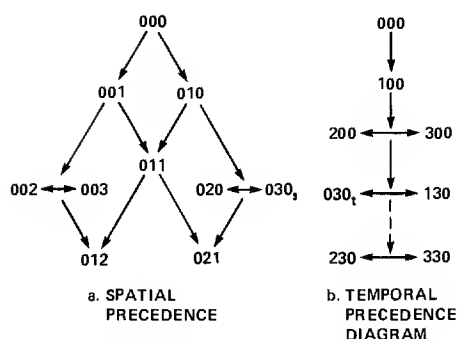


Fig. 7. Precedence diagrams.

to the static component of V_{030} ; if a fair amount of motion occurred, V_{030} would be subject to temporal precedence relations also. As the eye does not see spatial detail in areas of high motion, V_{030} may be included in a vector set designed only for temporal fidelity without its spatial predecessors.

An example of the types of distortion that can occur in "improper" vector sets is given in Fig. 8. Figure 8(a) shows a subpicture depicting a vertical edge; Fig. 8(b) lists the Hadamard coefficients in the subpicture; Fig. 8(c) shows the subpicture's reconstruction without V_{002} and V_{003} ; and Fig. 8(d) shows its reconstruction without V_{003} . While the edge blur is less in Fig. 8(d) than Fig. 8(c), this is more than offset by the extra dark bar in the right-hand side of the subpicture in Fig. 8(d). This causes an outline to appear around the picture; the outline is quite visible as the eye detects relatively low levels of patterned noise. Reconstruction with all vectors would yield the original subpicture shown in Fig. 8(a).

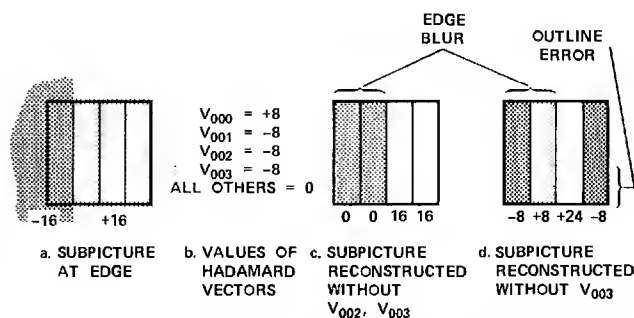


Fig. 8. Spatial ghosting effects.

A temporal precedence relation is given in Fig. 7(b). This vector ordering was developed and verified by observation of an oscilloscope displaying a horizontal line moving vertically at various speeds. This display was also viewed with the line at a 45° angle and vertically with motion being perpendicular to the line. A moving Lissajou pattern, with a frequency ratio of nearly 1:1, was also used. Smoothness of motion and lack of "ghosting" were the criteria used to judge the vector sets. The results were further checked by observing off-the-air video. The line between $V_{030} - V_{130}$ and $V_{230} - V_{330}$ is dashed to indicate that violation of this precedence produces only minor errors. In general, violation of precedence closer to V_{000} results in more severe errors in either of the diagrams. Only the more significant vectors have been listed in the temporal diagram. Also, the two diagrams can be combined into an overall precedence diagram, and this is a bit complex for this treatment.

The flashing "ghosts" in temporal vector sets not following this precedence relation arise from phenomena similar to the example given in Fig. 8. If the example is viewed with the horizontal direction representing time, and the vectors represented in Fig. 8(b) are V_{000} , V_{100} , V_{200} , and V_{300} , respectively, the right-hand bar in Fig. 8(d) becomes a ghost that flashes black in a white background.

Cutpoint and Representative Value Distributions

Previous research has shown that logarithmic quantization schemes fit the probability distributions of Hadamard vectors fairly well. Landau and Slepian,⁴ using a 4×4 transform, logarithmically quantized still pictures to 2 bits/pel. The Q's in Fig. 2(a) correspond to the bit assignments for each vector. Some experimentation was done at Ames to see if other nonlinear schemes would improve picture quality, but no significant improvement was discernible. The logarithmic quantization was therefore adopted. As all values quantized to the same number of bits are not identical in their probability distributions, the logarithmic quantization have been scaled to fit. As an example, in Table 1 below, the notation 4-1.3 means that the cutpoints and representative values are for a 4-bit quantization and scaled up by a factor of 1.3 over the figures given for 4 bits by Landau and Slepian. Cutpoints and representative values are given below in octal notation, as they would be entered in the programmer.

Table 1. Cutpoint and Representative Value Distributions

	1 2 5 12 21 32 43	(±cutpoints)		3 7 17
4-1.3	0 1 4 10 15 25 35 47	(±representative values)	3-10	0 5 12 24
	2 4 6 10 13 16 21			2 5 14
4-0.5L	0 3 5 7 11 14 17 23		3-1.3	0 3 10 17
	1 2 4 6 10 13 16			2 4 11
4-0.5	0 1 3 5 7 11 14 17		3-1.0	0 3 6 14
	2 3 4 6 10 13 16			2 3 7
4-0.5m	0 2 3 5 7 11 14 17		3-0.8	0 2 5 12
	1 2 3 4 5 6 10			3
4-0.3	0 1 2 3 4 5 7 12		2	0 6

Sample Algorithms

Table 2 lists vectors, subscripts, the number of bits they are quantized to, and the quantization. For the representative values and cutpoints, see Table 1.

Adaptive Algorithms

In the adaptive mode, the processor monitors some of the vectors to determine if the subpicture contains movement. Option zero is programmed with a vector set that gives high spatial fidelity and little or no temporal fidelity. Any temporal fidelity used in option zero is usually set to cover slow motion only, that is, only vectors with a temporal subscript of 1 are used.

The adaptive mode may employ two, three, or four options. The highest option used (1, 2, or 3) will be programmed to display high temporal, but low spatial fidelity. Intermediate options, if any, will trade off spatial and temporal resolution.

The processor can monitor up to four vectors. Three have been used in experiments so far: V₁₀₀, V₂₀₀, and V₃₀₀. The absolute values of each vector are compared to three cutpoints entered in ascending order on dip switches. These correspond to making a selection from four options. If fewer options are used (three or two) then fewer cutpoints are needed (two or one). If the magnitude of V₁₀₀, for example, falls

below the smallest cutpoint, it "votes" for option zero. If it falls between the first two, it "votes" for option 1 and so on. The vectors are thresholded independently, each vector having its individual set of cutpoints. The output of the thresholding section is a 0, 1, 2, or 3 from each vector.

The option finally selected by the processor is simply the maximum of these numbers. The reason for this is that motion that is very close to the pattern of one of the sending vectors will not produce large values of the other vectors (Hadamard vectors are orthogonal). Sometimes only one motion vector will have a high value in the midst of a rapidly moving area.

Motion Detection

The special case of a horizontally moving vertical edge serves to illustrate the behavior of temporal vectors, and how they can be used to detect motion. Figure 9(a) shows a picture of a black bar moving rapidly across a white background. Figure 9(b) shows a "freeze" of four frames. The blur of the moving area is shown by the arrows in Fig. 9(b) and Fig. 9(c), and is designated as "motion in 4/30 second" and "the extent of motion in 4 frames (8 fields)." Figures 9(b) and 9(c) can also be thought of as the time average of the 4 frames (8 fields). Figure 9(c) shows 4x4 subpictures in various parts of the picture. The numbers inside the squares give the number of black fields in a subpicture at that location. To the left of the moving area all fields are black and this number is 8; to the right, all are white, and the number is 0. Inside the moving area, fields in a given subpicture begin white and turn black as the edge crosses that subpicture. This field structure is shown in Fig. 9(d).

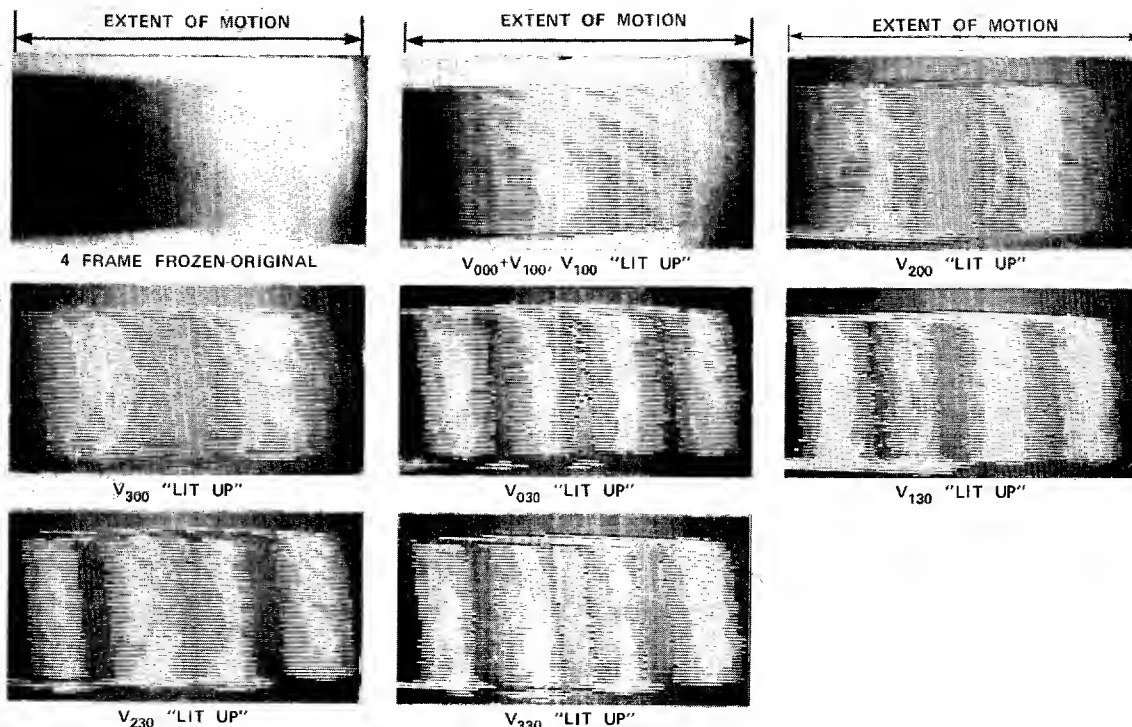
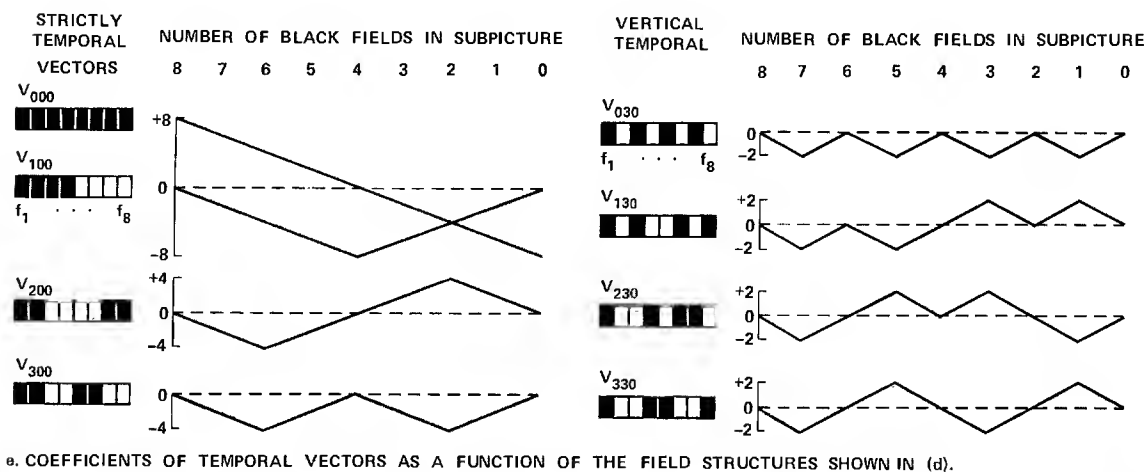
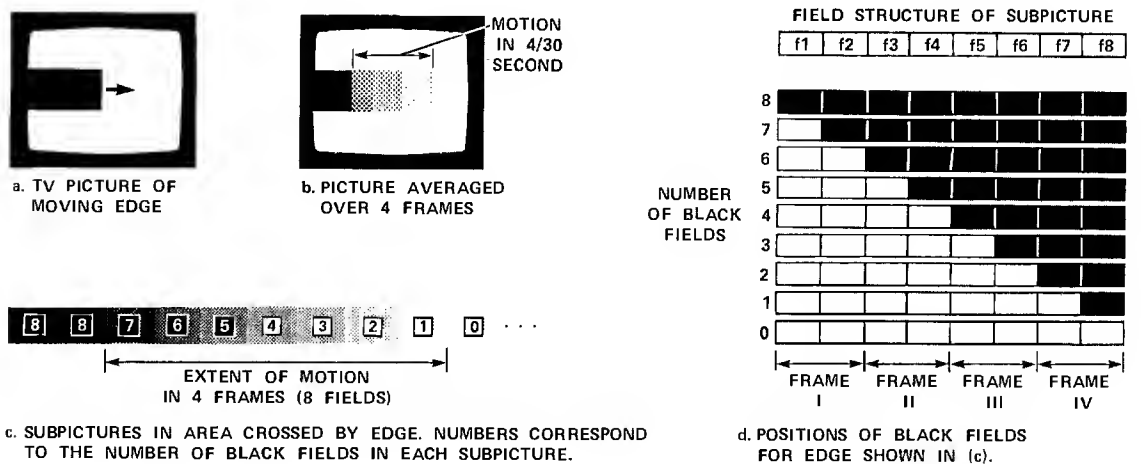


Fig. 9. Motion detection with temporal vectors.

The values of the coefficients of the major temporal and vertical-temporal vectors for these subpictures are shown in Fig. 9(e). The subscript notation is from the 4×4 subpicture model; however, the diagram under each vector shows the pattern of light and dark fields that the vector represents. A subpicture matching one of these vector patterns would give the maximum coefficient value for that vector, and zero for all others.

Figure 9(f) shows photographs taken of the output of the system illustrating the graphs in Fig. 9(e). Occurrences of each vector were mapped into V_{100} , and quantized with extra large representative values, to give the bar pattern of varying intensities. By looking at the edges, it can be observed that the bars are black at the top for positive values of the vectors being displayed and white at the top for negative values.

In this simple case, V_{100} and V_{200} would be adequate for detecting motion, as the magnitudes of the coefficients of V_{200} and V_{300} are equal. Moving lines, stripes, rotating objects, etc., are more complex, and V_{300} is required for these more general cases.

Example of an Adaptive Algorithm. A three-option adaptive algorithm that requires 1 bit/pel is described in Tables 3 and 4. The overhead for selecting the option is 2 bits/subpicture, leaving a maximum of 62 bits/subpicture to spend on each option. Good static fidelity requires only 32 bits - the cost of the spatial vectors in option zero. The 25 other bits are spent for slow motion on vectors with temporal subscripts of 1. These represent about 15 frames/sec of motion. Option 1 spends 6 fewer bits on static spatial fidelity and 4 fewer bits on slow moving spatial fidelity (V_{102} and V_{103} are dropped). With V_{200} and V_{300} , option 1 includes low ranges of all vectors required for 30 frames/sec of motion. The temporal components of V_{030} and V_{130} add low ranges of 60-field/sec motion capability.

The quantization of motion vectors in options 0 and 1 uses low ranges as these options are not selected for fast motion. Quantization curves, like 4-0.3 and 4-0.5, do a smooth job of quantizing the lower ranges of V_{100} , V_{200} , V_{300} , V_{030} , and V_{130} .

The high motion option provides some spatial resolution in V_{001} , V_{010} , V_{101} , and V_{110} , but concentrates most of its 61 bits on high fidelity representation of the principal temporal and vertical-temporal vectors.

Table 3. Vectors Programmed in a 3-Option Adaptive Algorithm

Option 0 - Static and Low Motion									
000	7	001	4-1.3	002	3-1.0	003	3-1.0		
010	4-1.3	011	3-1.0	012	3-0.8				
020	3-0.8	021	2	030	4-0.5m				
100	4-0.3	101	4-0.5	102	2	103	2		
110	3-0.8	130	4-0.5						
36 bits static, 61 bits total									
Option 1 - Medium Motion									
000	7	001	4-1.3	002	3-1.0	003	3-1.0		
010	4-1.3	011	2	020	3-0.8	030	4-0.5m		
100	4-0.5L	101	4-0.5	110	3-0.8	130	4-0.5m		
200	4-0.3	201	3-0.8	210	2				
300	4-0.3	301	2	310	2				
30 bits static, 62 bits total									
Option 2 - High Motion									
000	7	001	4-1.3	010	4-1.3	030	4-0.5m		
100	7	101	4-0.5	110	3-0.8	130	4-0.5m		
200	7	230	4-0.5m	300	7	330	4-0.5m		
15 bits static, 59 bits total									

Adaptive parameters are given in Table 4. Note that V_{200} and V_{300} never vote for option 1 as the two cutpoints are identical. This is done to reserve option 1 for medium motion of about 15 frames/sec, which corresponds to activity in V_{100} . Low amplitudes of higher frequency motion in option 1 are provided for by V_{200} , V_{300} , V_{130} , and V_{030} .

An illustration of the adaptive process at work for this algorithm is given in Fig. 10. The pictures in each sequence are: (1) 4 frames of digitized video - no compression; (2) 4 frames of video transformed with this algorithm; (3) same as (2), except that those areas selecting option 2 are shown as checkerboards; and (4) same as (2) with option 1 areas as checkerboards. The left-hand sequence shows a weatherman moving his hand on a map; the right-hand sequence shows Mr. Spock walking rapidly to the right of Dr. McCoy in 4 frames of Star Trek. The top picture in the latter series clearly shows the field structure, several individual "still" scenes - fields - are visible in the "blur" of motion.

The algorithms described in this paper have undergone preliminary subjective evaluation at Stanford University and are scheduled for more. Formal subjective evaluation is necessary to get a more accurate measure of picture quality; mean square error and other analytical measures do not model the complex motion detection functions of the eye. Obviously, a lot of informal observation and evaluation on the part of Ames personnel, using off-the-air programming and Stanford classroom video, was required in order to develop these algorithms.

Table 4. Adaptive Motion Sensing Parameters

Adaptive thresholds for V_{100} , V_{200} , and V_{300} :			
Cutpoints:	3 24	Cutpoints:	5 5
Option		Option	
voted:	0 1 2	voted:	0 1 2
V_{100}		V_{200} and V_{300}	

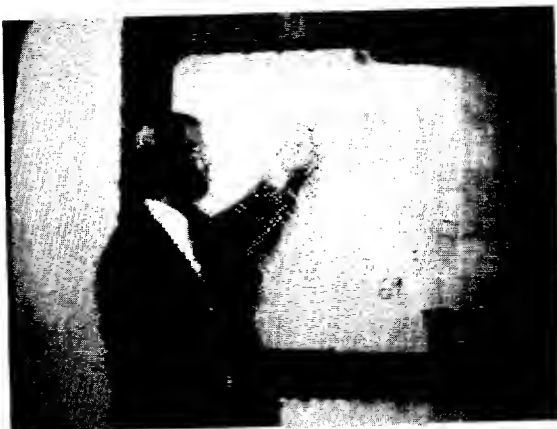
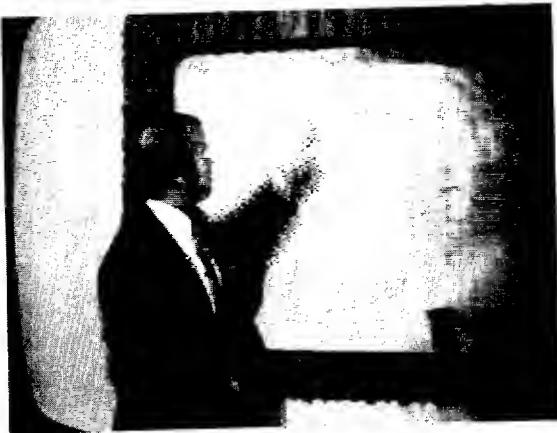
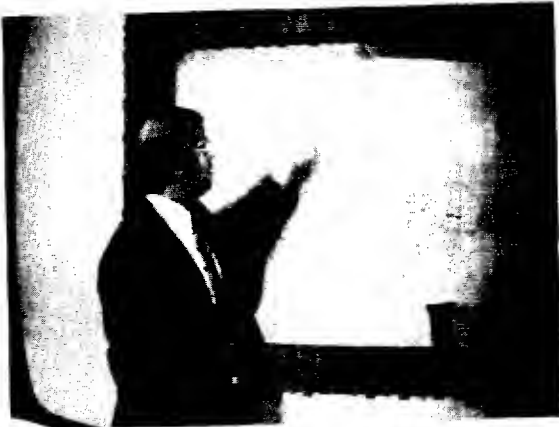


Fig. 10. Illustrations of adaptive mode selections.

Concluding Remarks

Algorithms for video compression with three-dimensional Hadamard transforms are not extensions of those for still pictures. The variance of a vector's probability distribution does not drop rapidly with increasing temporal sequency; video is a source with a lot of movement in four frames. The interlaced scan makes some vectors a mix of vertical and temporal information. Logarithmic quantization does work well for three-dimensional Hadamard vectors, however. In order that vectors add resolution, and not noise, they should be assembled in sets following a precedence relation.

Vectors that have both high spatial and high temporal sequency are not needed. This reflects the fact that the eye does not see movement and high detail simultaneously. This property of the eye can be further exploited by adaptive algorithms that segregate high spatial and high temporal sequency vectors in different options devoted to spatial or temporal resolution. Successful adaptation on the basis of movement can be done by monitoring only a few temporal vectors.

Further work involves extension to color, continued development of a compact processor to implement specific adaptive algorithms serially, investigation of a rate buffered, four-option adaptive algorithm in which the static and low-motion options have fewer bits per subpicture than the medium- and high-motion options, and more research to correlate motion vectors with specific types of motion.

References

1. Noble, S. C., Knauer, S. C., and Gien, J. I., "A Real-Time Hadamard Transform System for Special and Temporal Redundancy Reduction in Television," Proceedings of the International Telemetering Conference, October 1973, Washington, D. C.
2. Harmuth, H. F., *Transmission of Information by Orthogonal Functions*, Springer-Verlag, pp. 22-33, 42.
3. Pratt, W. K., Kane, J., and Andrews, H. C., "Hadamard Transform Image Coding," *Proc. IEEE*, Vol. 57, No. 1, Jan. 1969.
4. Landau, H. J., and Slepian, D., "Some Computer Experiments in Picture Processing for Bandwidth Reduction," *Bell System Technical Journal*, Vol. 50, pp. 1525-1540, May-June 1971.

AN ADVANCED IMAGING COMMUNICATION SYSTEM FOR
PLANETARY EXPLORATION*Robert F. Rice
Jet Propulsion Laboratory
Pasadena, California 91103Abstract

An advanced imaging communication system (AICS) for planetary exploration is presented. The system offers "end-to-end" information rate improvements of 3 to 5 times over existing systems in addition to extensive user flexibility to adapt his rate/fidelity priorities to fit a particular mission. AICS contains two major system elements. The first is a concatenated Reed-Solomon/Viterbi coded channel. This provides a powerful, yet practical, solution to the usual "error vulnerability" problem associated with compressed data. The second major element is an extremely adaptive image data compression algorithm called RM2. The details of this algorithm as presently simulated are discussed in considerable detail. Used in conjunction with the virtually error free performance of the Reed-Solomon/Viterbi channel the stated AICS advantages are obtained.

INTRODUCTION

The development of communication systems for deep space imaging applications has historically proceeded with no real consideration for the system implications of data compression. Data compression to be useful on imaging or non-imaging data sources, would have to exhibit significant rate/quality advantages after transmission over channels designed for uncompressed data. To correct for the characteristic error sensitivity of compressed data, the familiar bit error rate/communication rate tradeoff often promised to eliminate many of the potential advantages data compression offered. One element of an Advanced Imaging Communication System (AICS) discussed here solves this classic problem in a straightforward and practical way, making direct use of a current investment in coded communications. This aspect of AICS is extensively documented in Refs. 1-3 and will therefore receive only a summary treatment in the introductory portions of this paper. Making use of these rather convincing results, the remainder of this paper concentrates on the source coding problem without the usual fear of channel errors. In particular, a detailed description of a new imaging data compression algorithm (RM2) and associated system considerations are given.

Jupiter/Saturn Communication System^[1]

The communication system being implemented for a Mariner Jupiter/Saturn flyby mission in 1977 is shown in Fig. 1. The capabilities of this system are, of course, intended to be available for any Mariner or Pioneer mission in the post 1977 period. Representing an advance in communication efficiency, we will first review its characteristics and later compare them to those of AICS.

The principal investment in this system is the introduction of a high rate coded channel, the elements of which have been labeled collectively as the Jupiter/Saturn Viterbi Channel. This includes an onboard convolutional coder with constraint length $K = 7$ and code rates of $1/2$ and $1/3$, the usual modulation/demodulation techniques with X-band and S-band carriers, Viterbi decoders (with 3 bits of receiver quantization) located at several Deep Space Network receiving stations, followed by ground communication links. [1], [4], [5]

Non-imaging. Although general science and engineering data (GSE) have typically represented less than 20% of the available data rate during planetary encounters, it is unrealistic to exclude it from the overall communication system trade-offs. It is therefore relevant to our principal

* This paper presents the results of one phase of research carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract No. NAS7-100, sponsored by the National Aeronautics and Space Administration.

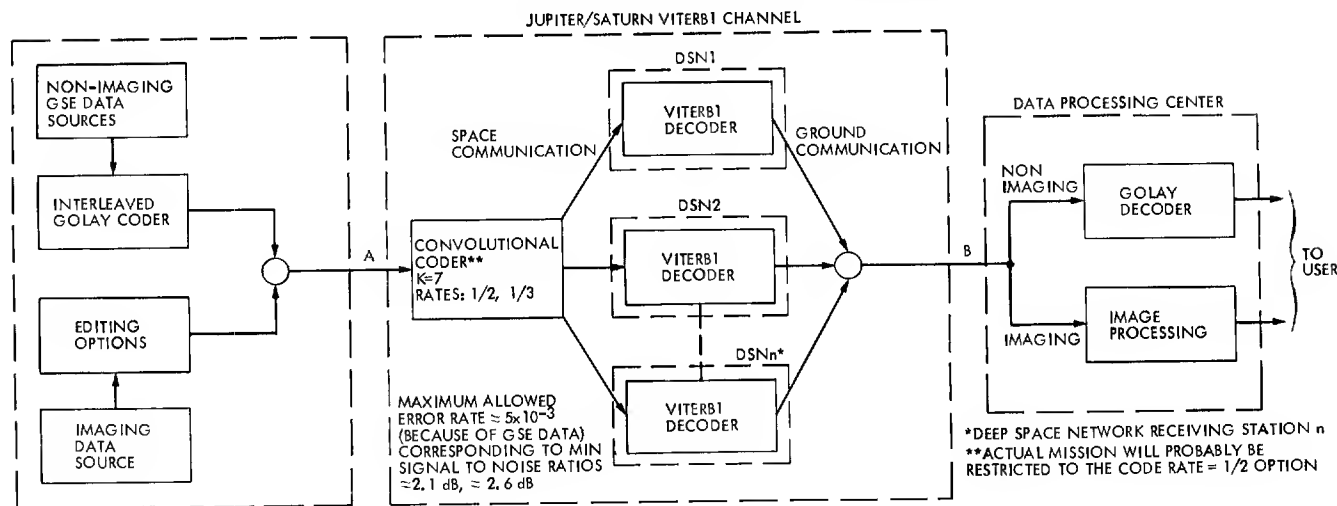


Fig. 1. Jupiter/Saturn Communication System.

endeavor here to communicate imaging data more efficiently.

Imaging data in uncompressed, sampled and quantized form is relatively insensitive to communication errors compared to GSE data. Whereas bit error rates of 5×10^{-3} or so may be acceptable for uncompressed imaging, error rates closer to 5×10^{-5} are desired for some forms of GSE data. This disparity was accommodated in the uncoded communication system of the recent Mariner Venus/Mercury mission by providing two separate channels. The Jupiter/Saturn Communication System provides an acceptable single channel solution by operating the Jupiter/Saturn Viterbi Channel at around 5×10^{-3} and relying on additional coding of the GSE data to lower the GSE error rate to an acceptable level. The additional coding took the form of an interleaved Golay block coder/decoder as shown in Fig. 1. Because only half of the bits in a Golay (modified to (24, 12)) codeword represent information, the effective data rate of GSE data through the "inner" Jupiter/Saturn Viterbi Channel is reduced by a factor of two. This is an acceptable inefficiency provided GSE data represents only a small fraction of the total.

Operation of the Jupiter/Saturn Viterbi Channel at the bit error rate of 5×10^{-3} occurs at signal to noise ratios of about 2.1 dB for the code rate = 1/3 option and about 2.6 dB for the rate = 1/2 option. When Golay coded GSE data is

communicated, operation appreciably below these operating points (i. e., lower signal to noise ratio, higher transmission rate, higher error rate) is not possible because of severe degradation to GSE data.

Imaging. Easily implemented imaging rate options are provided by allowing 2-dimensional on-board editing of picture elements (pixels) and using linear interpolation for ground reconstruction. Such options are acceptable because at bit error rates of 5×10^{-3} (2.1 dB and 2.6 dB) and less, pixel edited data is not significantly more sensitive to errors than standard PCM.

Data compression. The above communication system offers some impressive advantages over previous systems. For example, at the same signal to noise ratio that achieves the desired 5×10^{-3} bit error rate (2.1 dB) with the $K = 7$, code rate = 1/3 option, an uncoded system would have a 1/33 bit error rate (e.g., the recent Mariner Venus/Mercury flybys). However, the system suffers the usual difficulties when used to communicate error sensitive compressed data.

Under ideal phase coherent receiver operating conditions a 3 dB rate reduction is required to reduce the bit error rate of the Jupiter/Saturn Viterbi Channel from 5×10^{-3} to about 10^{-6} . Under non-ideal conditions the loss is much more significant with some question of whether such low bit error rates can even be achieved. To be

able to use any data compression algorithm on this channel thus means an initial rate reduction of at least 3 dB. In general each algorithm is affected in its own unique way by channel errors, resulting in a continuous rate/fidelity tradeoff. Given the task of selecting from a set of imaging data compression algorithms to use on this channel would justifiably lead to extensive simulations, analysis and most likely inconclusive results. In practice such studies tend to obscure the real source coding problem and make ambiguous the potential system advantages of data compression. AICS offers a practical solution to these difficulties.

AICS

A block diagram of AICS is given in Fig. 2. As shown, the center box representing the Jupiter/Saturn Viterbi Channel in Fig. 1 remains unchanged in AICS. The Golay coding has been eliminated for GSE data and the editing options for imaging have been replaced by a data compression algorithm labeled RM2. The elements which remedy the channel problem is an interleaved Reed-Solomon block coder and corresponding decoder located at a destination data processing center.

The channel. [1]-[3] All source data in an AICS system is communicated through the concatenated Reed-Solomon/Viterbi channel, C-D in Fig. 2. Under ideal receiver operating conditions, the channel C-D will exhibit virtually error free performance at approximately the same signal to noise

ratio (2.3 dB and 2.8 dB between C and D for code rates 1/3 and 1/2) as that required by the Jupiter/Saturn Viterbi Channel, A-B, to achieve a 5×10^{-3} bit error rate. For the rate 1/3 option this is only a little more than 4 dB from the channel capacity of the infinite bandwidth white gaussian noise channel.

The concatenated system performance curves are so steep (an order of magnitude change in error rate per 0.1 dB change in signal to noise ratio) that the usual tradeoffs are of no consequence. This all means that imaging data in an AICS system, compressed or not, can be transmitted at the same maximum communication rate of uncompressed imaging data in a Jupiter/Saturn Communication System, Fig. 1. As an added bonus, the effective GSE data rate is substantially increased during both encounter and cruise periods, and burst error protection of over 2000 bits is provided.

Unlike practical sequential decoding systems, the concatenated system is extremely "robust" in the face of receiver imperfections such as AGC and phase tracking. Losses of the concatenated system under nearly error free conditions (i. e., a burst of errors once every 100 pictures or so) is no more than that of the "inner" Viterbi system operating at a relatively high 5×10^{-3} bit error rate.

Because parity symbols in a properly selected Reed-Solomon code contribute less than 15% in overhead, ground communications within the Jupiter/Saturn Viterbi Channel (see Fig. 1)

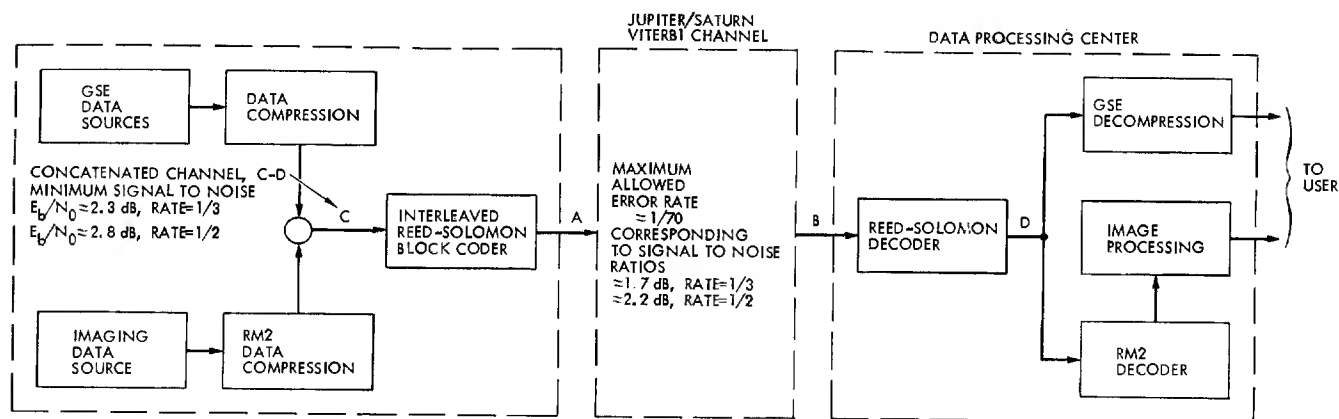


Fig. 2. Advanced Imaging Communication System (AICS).

are unaffected. This permits a single decoder to be located at a central location. With hardware estimates of 150 off-the-shelf integrated circuit chips for a 100 KBS interleaved Reed-Solomon decoder and 60 or so for a coder (in addition to the elimination of the Golay) this performance is obtained at a small increase in overall system complexity. The system is thus highly practical.

For those desiring more details, Chapter 4 of Ref. 1 contains extensive system considerations leading to these conclusions. It is an extension of Odewald's original work in [2] to include primarily the interaction of the source with the channel in addition to imperfect receiver operating conditions, interleaved code block synchronization, etc. This led to another study^[3] with Linkabit Corp. in which performance projections in [1] were verified by direct simulation.

Before turning exclusively to the source coding problem, it is of interest to consider the structure in Fig. 2 from another viewpoint. Instead of treating the Reed-Solomon coding/decoding as part of a concatenated channel it can instead be considered part of the source coding/decoding with compression factors adjusted down by the slight 15% parity overhead. The two outer dotted boxes become source coders and source decoders. Then, provided the average bit error rate on the Jupiter/Saturn Viterbi Channel is less than about 1/70, compressed data (i. e., including the Reed-Solomon mapping) is unaffected.

Data compression. The now practical noiseless channel model will be assumed for the remainder of this paper. Rate/fidelity tradeoffs involving purely source coding concepts can be considered as true "end-to-end" communication system tradeoffs. The benefits derived may similarly be considered as end-to-end benefits.

The usual source coding problem of finding the best way to code an image with a prescribed number of bits is only a subset of the source coding problem considered in the AICS development. The bigger problem, loosely stated, involves seeking a means of giving the user the capability to distribute a fixed number of bits over the surface of a planet(s) in a way which maximizes his scientific information return (imaging and non-imaging).

Although such a problem will hardly yield to rigorous theoretical analysis, simply considering the obvious role of the scientific user in planning imaging sequences leads to a powerful and practical source coding system structure.

The basic philosophy behind the AICS source coding system structure is to provide an extreme range of adaptivity at all levels of coding. At the level of user intervention, this means giving the user extensive capability to tradeoff rate and fidelity within an imaging sequence. Discussions will start with this global view in which a new imaging data compression algorithm, called RM2, is treated as a user controlled system element. The development proceeds naturally to providing most of the detailed structure of RM2 as presently simulated. Some results of these simulations follow.

Input buffer capability of at least a full image should be assumed throughout the discussions. This is a quite practical assumption today. A CCD memory structure is currently planned for advanced Pioneer missions incorporating imaging and additional technologies are being considered for future Mariners.

CODING IMAGE SEQUENCES

As an element in a system which codes user planned sequences of images an adequate initial description of RM2 is simply a rate controlled data compressor. Given an input "aim" rate of B_{AIM} bits/picture, RM2 will seek to use precisely B_{AIM} bits to code an image. The actual bits used, B_{ACT} , will be a close approximation to B_{AIM} . A user's choice of B_{AIM} is fundamentally unlimited but is practically limited to values above about 0.25 when expressed in bits per picture element (bits/pixel). Reproduced quality for a given image is monotonically related to $B_{ACT} \approx B_{AIM}$. Simulation results have demonstrated consistently excellent rate/quality performance on a wide class of representative images. The latter statement assumes a fixed set of RM2 internal parameters.

A user planned imaging sequence spanning a time period of τ seconds must satisfy the equations.

$$B_i^I(0) + B_i^{GSE}(0) = r \cdot \tau_i \quad (1)$$

for the i th image and

$$\sum_{i=1}^N \tau_i = \tau \quad (2)$$

where

$r \equiv$ communication rate in bits/sec

$\tau_i \equiv$ allotted communication time i th image

$B_i^I(0) \equiv$ user allocated bits to i th image

$B_i^{GSE}(0) \equiv$ user allocated bits to GSE data during i th image time

The wide range of RM2 performance options gives the user a practical tool for accomplishing a difficult source coding problem. By choosing N and assigning the rate allocations $\{B_i^I(0)\}$ and $\{B_i^{GSE}(0)\}$ a scientific user is, in effect, trying to place $r\tau$ bits on the target in a manner which maximizes his expected information return. Each $B_i^I(0)$ is his assessment of the "relative" importance of the i th image out of all images he might take in time τ , weighted by his a priori knowledge of RM2 performance characteristics.

The average bit rate assigned to GSE data, $B_i^{GSE}(0)/\tau_i$, would typically be constant and only a fraction of the total bit rate, (e.g., <0.3). However, it is desirable to be able to accommodate short term variations in actual GSE data rate as the result of variability in non-imaging data compression algorithms (see Fig. 2) and/or the detection of important scientific events which simply require more bits. To accommodate the latter and the fact that RM2 itself doesn't always use precisely its allocated bits we incorporate a simple control system.

Rate Control Loop

The simple control system showing both RM2 and GSE coding is given in Fig. 3. Starting at the input, the bit assignment to the i th image $B_i^I(0)$ is adjusted by the amount λ_i (zero at start of sequence) giving RM2 an aim rate of B_i^I bits. The i th image is coded to $\hat{B}_i^I \approx B_i^I$ bits. "Unused" image bits, $B_i^I - \hat{B}_i^I$, are added to unused GSE bits yielding ϵ_i , which is added to unused bits from previous images, E_i . Rate adjustments for image

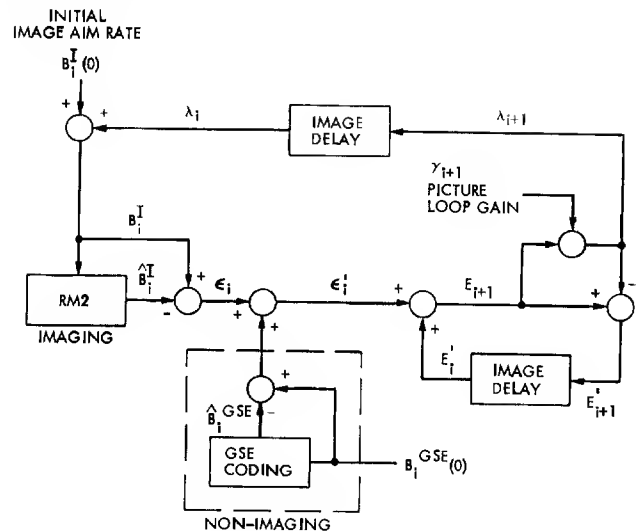


Fig. 3. Rate Control Loop, Image Sequence.

$i + 1$ is computed as $\gamma_{i+1} E_{i+1}$, a fraction of all unused bits up to picture $i + 1$. γ_{i+1} can be thought of as the loop gain for picture $i + 1$. E_{i+1} can be viewed as the negative of a buffer fullness marker (about some midpoint).

As noted earlier, variations in GSE coded data rate from a user planned $B_i^{GSE}(0)$ allocation is presumed to be primarily the result of onboard detection of important scientific events or variability of data compression algorithms. The configuration in Fig. 3 allows for this variability to be distributed over subsequent images depending on the loop gains, γ_i . Suppose for example that for all i $B_i^I(0) = 4/5 B$, $B_i^{GSE}(0) = B/5$ and a burst of information caused \hat{B}_i^{GSE} to be 50% higher than the allowed B_i^{GSE} . With a loop gain of $1/5$ the control loop would distribute 20% of the error, $B/10$, into the $i + 1$ st image. Assuming no error up to the i th, B_{i+1}^I would be modified from $4/5 B$ to $39/50 B$, an almost imperceptible change of 2%.

Last minute minor changes in GSE data priorities or fine adjustments in the communication link capability, r , could be handled by modifying all $B_i^I(0)$ to again satisfy (1) or by treating the change as a step input in Fig. 3.

Simulation results of RM2 have thus far shown that, for images considered to be of primary interest, the error $|\epsilon_i|$ is typically less than 1% of B_i^I . Without the need to gracefully accommodate perturbations, such as described above, γ_i could be made equal to unity for all i , simplifying Fig. 3.

IMAGE CODING

The internal structure of RM2 looks much like that in Fig. 3 with only the slight relabeling shown in Fig. 4. Images are treated as sequences of subpictures (arrays of pixels) in a manner similar to the way sequences of images are treated in the overall system. RM2 in Fig. 3 is replaced by an "array processor" which, given an input bit allocation b_k for the k^{th} array, will code that array to $\hat{b}_k \approx b_k$ bits. Given an allocation of B_i^I bits for the i^{th} image (Fig. 3) the initial bit allocations, $b_k(0)$, to each of n arrays must satisfy the constraint

$$\sum_{k=1}^n b_k(0) = B_i^I \quad (3)$$

This is similar to the constraints in (1) and (2) except now the determination of individual rate allocations can no longer be provided by user intervention. The control loop in Fig. 4 handles rate errors, $|b_k - \hat{b}_k|$, in a manner similar to that in Fig. 3. Simulation results indicate that an array loop gain of about 0.35 appears to be entirely satisfactory.

Rate Allocation

Before actual coding of an image, RM2 first obtains features from each subpicture (array) which are useful in establishing priorities for bits throughout the picture.

In some circumstances (e.g., viewing satellites) many arrays may be entirely "black space" and carry no useful information. Simple detection schemes can easily determine when an array should be treated as such.

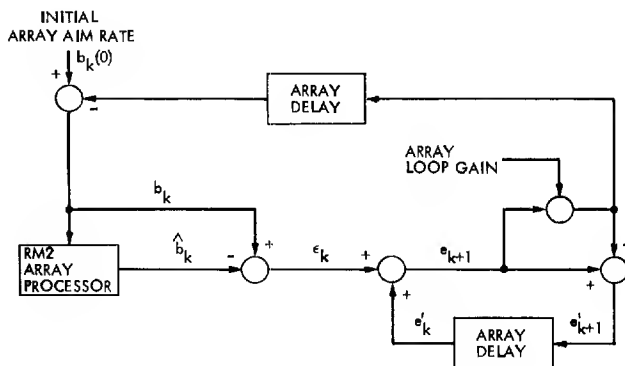


Fig. 4. Rate Control Loop, Array Sequence

Let

$$BS_k \equiv \begin{cases} 0 & \text{If } k^{\text{th}} \text{ array determined to be black space} \\ 1 & \text{Otherwise} \end{cases} \quad (4)$$

Using $*$ to denote concatenation, the n -bit header $BS = BS_1 * BS_2 * \dots * BS_n$, appearing at the start of a coded image, would specify which arrays would be missing in subsequent coded information. This leaves $B_i - n$ bits to be distributed among

$$n' \equiv \sum_k BS_k \quad (5)$$

arrays in the i^{th} image.

For those arrays which are not "edited" in (4) a feature we call "normalized activity" is computed. It is basically an estimate of the bits required to code an array under a prescribed set of approximation parameters, normalized so that the smallest equals unity. In particular, let A'_{kj} be the normalized activity of the j^{th} array which has not been edited, where $j = 1, 2, \dots, n'$.

Define M monotonically increasing "enhancement constants" $e(m)$, $m = 1, 2, \dots, M$ and the constant α_{MIN} . Then, starting with $m = 1$ and increasing in order, determine

$$\text{SUM}(m) \equiv \sum_{j=1}^{n'} (A'_{kj})^{e(m)} \quad (6)$$

and

$$\alpha_{kj}^m \equiv \frac{(A'_{kj})^{e(m)}}{\text{SUM}(m)} \quad (7)$$

Let m^* be the largest m such that

$$\min_j \alpha_{kj}^{m^*} \geq \alpha_{\text{MIN}} \quad (8)$$

The initial bit allocation for array k_j of picture i is determined by

$$b_{kj}(0) = \alpha_{kj}^{m^*} \cdot B_i \quad (9)$$

where B_i is the bit allocation to the i^{th} image (Figs. 3 and 4).

Letting p be the number of pixels in an array, Equa. 9 may be written in a more recognizable form as

$$b_{k_j}(0)/p = \left(n' \alpha_{k_j}^{m*} \right) \frac{B_i}{n'p} \quad (10)$$

The lefthand side of (10) is the initial rate assignment to array k_j in bits/pixel. The term $B_i/n'p$ represents a uniform (equal) rate assignment, in bits/pixel, to all arrays that haven't been edited. Thus $n' \alpha_{k_j}^{m*}$ can be viewed as a multiplier which modifies a rate assignment from the uniform one. A user would specify $n' \alpha_{\text{MIN}}$, rather than α_{MIN} , for a sequence or mission.

Observe that if an image is uniform in activity all the $\alpha_{k_j}^m$ in (7) will equal $1/n'$ so that the multiplier $n' \alpha_{k_j}^{m*}$ in (10) is unity for all arrays. A uniform distribution of rate results. By properly choosing the $e(m)$, the smallest multiplier will gradually decrease to a limiting value of $n' \alpha_{\text{MIN}}$ as the spread in activity increases. Simulations have shown that selecting a good fixed set of $e(m)$ is an uncritical procedure. A rate allocation example is provided later in the section titled "SIMULATIONS."

ARRAY CODING

Transform

Basic to the coding of arrays in RM2 is a simple transform using cascaded two by two Hadamard transforms.^[6] We will be brief here.

Denote the array of data points by $C_0(0)$ where the zero in parenthesis means transform level zero. We will assume 64 by 64 pixel arrays, the same used in present simulations.

Any two by two array can be considered as a four component vector. Given two such arrays \tilde{A} and \tilde{B} , with components a_{ij} and b_{ij} respectively, the scalar product of \tilde{A} and \tilde{B} is given as

$$\langle \tilde{A}, \tilde{B} \rangle \equiv \sum a_{ij} b_{ij} \quad (11)$$

Given the four orthogonal basis vectors

$$\tilde{H}_0 \equiv \frac{1}{4} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad \tilde{H}_1 \equiv \frac{1}{4} \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \quad (12)$$

$$\tilde{H}_2 \equiv \frac{1}{4} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad \tilde{H}_3 \equiv \frac{1}{4} \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

any \tilde{A} can be written as

$$\tilde{A} = 4 \sum_k c_k \tilde{H}_k \quad (13)$$

where the

$$c_k = \langle \tilde{A}, \tilde{H}_k \rangle \quad (14)$$

are two by two Hadamard transform coefficients. Applying these operations on each two by two in data array $C_0(0)$ yields a 32 by 32 array of c_0 coefficients denoted $C_0(1)$. The collection of all other coefficients we denote by $C(1, 0, 1)$. The '1' in $C_0(1)$ and the first '1' in $C(1, 0, 1)$ stands for transform level 1. The meaning of the additional notation will become clear later. Given $C_0(1)$ and $C(1, 0, 1)$, the original data array $C_0(0)$ can be recovered by using each complete set of four coefficients in the inverse represented by (13).

The coefficients in $C(1, 0, 1)$ are not transformed further. However, the c_0 coefficients in $C_0(1)$, which are two by two averages, statistically behave much like the input data. The two by two operations in (13) and (14) are applied to $C_0(1)$ yielding a second level of transform coefficients, $C_0(2)$ and $C(2, 0, 1)$. In general, the procedure can be applied to $C_0(\ell)$ at transform level ℓ yielding $C_0(\ell + 1)$ and $C(\ell + 1, 0, 1)$. A block diagram of operations is shown in Fig. 5 where T is used to denote the collective transform operations on each two by two making up an input array.

Coefficient Approximation

In the present simulations four levels of transform coefficients are used although one or two

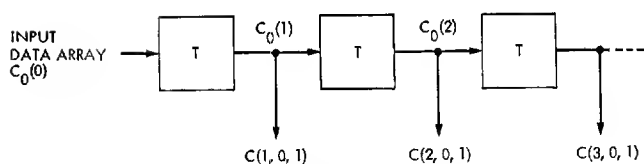


Fig. 5. Cascaded Transform

should be adequate in more restrictive applications. Starting with 8-bits/pixel input data the $C_0(\ell)$, $\ell \geq 1$ terms are maintained at a rounded 9 bits during the cascaded transform operations. Except for the highest level, four here, coded approximations to the $C(\ell, 0, 1)$ represent the transmitted data. We next define what these approximations are and how a particular set are chosen so that the total number of bits used is close to the allocated number (b_k for the k^{th} array in Fig. 4).

For our purposes all coefficients in $C(\ell, 0, 1)$ can be thought of as stored in memory as a 9-bit number plus sign (although a subtlety can reduce this to 7 bits plus sign). Any such coefficient c is represented in memory as the integer $4c$. Thus two of the ten bits are actually fractional. To avoid worrying about fractions

we define $C'(\ell, 0, 1)$ as the set of coefficients in $C(\ell, 0, 1)$, each multiplied by four. (15)

If $C_0(4)$ and each of the $C'(\ell, 0, 1)$ collections of stored coefficients were transmitted then the original data array, $C_0(0)$, could be reconstructed almost exactly by reversing the procedure in Fig. 6. An occasional error would occur due to roundoff errors during the transform process. The distribution of stored coefficients in $C'(\ell, 0, 1)$ tend to be distributed in the familiar unimodal form about zero. Such distributions are conducive to efficient variable length coding with the result that an almost perfect $C_0(0)$ could be communicated with between 3 to 6 bits/pixel. To achieve lower rates, down to as low as 0.1 bits/pixel, RM2 makes use of easily implemented entropy reducing approximations to the numbers in $C'(\ell, 0, 1)$ and to which efficient coding can also be applied.

Shift and round. A sequence of entropy reducing approximations is trivially obtained by simply

reducing the degree of linear quantization. To be explicit let $c'(0, 1) \equiv 4c(0, 1)$ be the stored integer representation of the true coefficient $c(0, 1)$. That is, $c'(0, 1) \in C'(\ell, 0, 1)$ and $c(0, 1) \in C(\ell, 0, 1)$. Define an approximation to $c'(0, 1)$ by

$$\begin{aligned} c'(s, 1) &\equiv S_s [c'(0, 1)] \\ &\equiv \frac{c'(0, 1)}{|c'(0, 1)|} \lceil 2^{-s} |c'(0, 1)| \rceil \\ &\quad + 0.5 \lceil 2^s, s \geq 0 \end{aligned} \quad (16)$$

where $\lceil \xi \rceil$ means the largest integer less than or equal to ξ . The corresponding quantized value of the true coefficient is just

$$c(s, 1) \equiv c'(s, 1)/4 \quad (17)$$

Increasing values of the "shift" parameter result in coarser quantization. At the same time, the entropies of the distributions of $c'(s, 1)$ that result decrease, taking on the same unimodal form but with gaps. Such gaps in the distributions of $c'(s, 1)$ when $s > 0$ can be removed for coding purposes by treating the integer numbers

$$2^{-s} c'(s, 1) = \frac{c'(0, 1)}{|c'(0, 1)|} \lceil 2^{-s} |c'(0, 1)| \rceil + 0.5 \quad (18)$$

instead of $c'(s, 1)$. This means simply leaving the shift in after the shift and round operations.

If $S_s[\cdot]$ is applied to all stored numbers in $C'(\ell, 0, 1)$ the resultant quantized set of numbers is denoted by $C'(\ell, s_\ell, 1)$. Again, an approximation to all the true coefficients in $C(\ell, 0, 1)$ is obtained by dividing all such numbers by four as in (17), yielding $C(\ell, s_\ell, 1)$.

Thresholding. If efficient coding were applied to the sets of numbers $C'(\ell, s_\ell, 1)$ the required coding rate in bits/number would decrease in steps as s_ℓ increased from zero. At the same time the fidelity of reproduced coefficients that result in $C(\ell, s_\ell, 1)$ would decrease. These

rate/fidelity options tend to be too coarse for our purposes here so that we need to define additional approximations to $C'(\ell, s_\ell, 1)$.

Given a positive or negative number ζ we define a thresholding operation with parameter $t \geq 1$ by

$$T_t[\zeta] \equiv \begin{cases} 0 & \text{if } |\zeta| < t \\ \zeta & \text{otherwise} \end{cases} \quad (19)$$

To obtain additional rate/fidelity options we apply $T_{t_\ell}[\cdot]$ to all numbers in $C'(\ell, s_\ell, 1)$ after removing the distribution gaps via (18). Specifically, with $c'(s_\ell, 1) \in C'(\ell, s_\ell, 1)$ the set of numbers

$$c'(s_\ell, t_\ell) \equiv 2^{s_\ell} T_{t_\ell} \left[2^{-s_\ell} c'(s_\ell, 1) \right] \quad (20)$$

will be denoted by $C'(\ell, s_\ell, t_\ell)$. As before the corresponding approximation to the original set of coefficients, given as $C(\ell, s_\ell, t_\ell)$, is obtained by dividing all numbers in $C'(\ell, s_\ell, t_\ell)$ by four as in (17).

Distribution gaps. The gaps in the distributions of numbers in $C'(\ell, s_\ell, t_\ell)$ are easily removed for coding purposes. The effect of quantization is removed by multiplying each element, $C'(s_\ell, t_\ell)$, by 2^{-s_ℓ} yielding from (20)

$$y(s_\ell, t_\ell) \equiv 2^{-s_\ell} c'(s_\ell, t_\ell) = T_{t_\ell} [2^{-s_\ell} c'(s_\ell, 1)] \quad (21)$$

and where we note that (18) is just a special case with $t_\ell = 1$. To remove the gaps caused by thresholding requires no more than the reversible transformation from $y(s_\ell, t_\ell)$ into $z(s_\ell, t_\ell)$ where

$$|z(s_\ell, t_\ell)| \equiv \begin{cases} 0 & \text{if } y(s_\ell, t_\ell) = 0 \\ |y(s_\ell, t_\ell)| - t_\ell + 1 & \text{Otherwise} \end{cases} \quad (22)$$

The resulting distributions of numbers will have no gaps caused by the operations $S_{s_\ell}[\cdot]$ and $T_{t_\ell}[\cdot]$. They can be characterized as unimodal with a spike at zero which increases with increasing t_ℓ .

A summary of the relationships described above is given in Fig. 6.

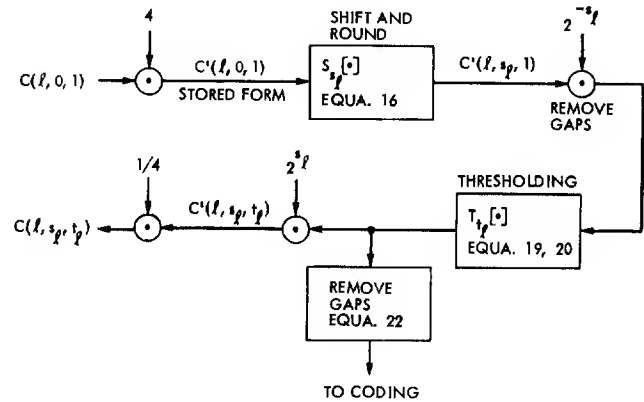


Fig. 6. Shifting and Thresholding.

Coding Definitions

Given a set of numbers, x , we let $\Phi[x]$ denote the coded form of x . That is, $\Phi[x]$ is a sequence of bits which can be "decoded" back into x . $\mathcal{L}(\Phi[x])$ will denote the length of sequence $\Phi[x]$ in bits. $\hat{\mathcal{L}}(\Phi[x])$ will mean an estimate of $\mathcal{L}(\Phi[x])$.

Space limitations here prevent a precise specification of $\Phi[\cdot]$, $\mathcal{L}(\cdot)$ and $\hat{\mathcal{L}}(\cdot)$ when applied to the various data sets such as $C'(\ell, s_\ell, t_\ell)$. Such details will appear in a later document in progress. However, the techniques basically involve extensions of the adaptive variable length coding concepts described in Ref. 7. Fortunately the simple definitions given above are adequate for our purpose here.

Interestingly, the variable length coding effectively carries RM2 rate allocation to a minute level, assigning more bits to local spatial areas with active data (e.g., transitions) than to regions which are relatively quiet.

Allowed s_ℓ, t_ℓ

In the present simulations the following data sets may be selected for communication at levels 1, 2 and 3

$$C'(\ell, 0, 1) \text{ and } C'(\ell, s_\ell, t_\ell) \begin{cases} s_\ell = 0, 1, 2, 3, 4 \\ t_\ell = 2, 3, 4, 5 \end{cases} \quad (23)$$

This gives 21 rate/fidelity options to choose from at each of the first three transform levels.

At level 4 no thresholding or quantizing is applied to the stored coefficients. Both $C(4, 0, 1)$ and $C_0(4)$ are encoded without modification. The choices in (23) reflect present simulations.

Coded Output Format

Let \bar{s}_ℓ and \bar{t}_ℓ be the chosen parameters at level ℓ , then a coded approximation to the original input array $C_0(0)$, denoted $\hat{C}_0(0)$, will appear as

$$\begin{aligned} \Phi[\hat{C}_0(0)] &= \Phi[C_0(4)] * \Phi[C'(4, 0, 1)] * \bar{s}_3 \\ &* \bar{t}_3 * \Phi[C'(3, \bar{s}_3, \bar{t}_3)] * \bar{s}_2 \\ &* \dots * \bar{t}_1 * \Phi[C'(1, \bar{s}_1, \bar{t}_1)] \end{aligned} \quad (24)$$

where again $*$ denotes concatenation and $\bar{s}_\ell, \bar{t}_\ell$ are assumed to be in binary form. The remaining problem is to specify how \bar{s}_ℓ and \bar{t}_ℓ are selected so that

$$\mathcal{L}(\Phi[\hat{C}_0(0)]) \approx b \quad (25)$$

where b is the bit allocation to the particular array before processing begins (b_k for the k th array in Fig. 4). Related to this problem is a definition of "array activity," the feature used in (6) - (10) to reallocate a fixed number of bits among several arrays.

Initial Rate Allocations

After coding transform level four there remains

$$b' \equiv b - \mathcal{L}(\Phi[C_0(4)]) - \mathcal{L}(\Phi[C'(4, 0, 1)]) \quad (26)$$

bits to distribute between levels 1, 2 and 3.

At this stage the problem becomes much the same as we have already seen in coding first sequences of images (Fig. 3) then sequences of arrays making up an image (Fig. 4). Here instead we have a sequence of transform levels, starting with level 3. The b' bits in (24) must be initially distributed between the three levels in a

manner which reflects their relative need for bits. Once the initial allocation of bits is completed, coding is initiated at level 3 and then proceeds on to level 2, then 1 (see 24). A control loop, structured exactly as in Figs. 3 and 4, takes care of adjusting the bit allocations at lower levels because of errors between the bits allocated and the bits used at higher levels. This procedure should be self-explanatory at this point and no diagram will be given.

Let $r_\ell(0)$, $\ell = 1, 2, 3$ be the allocated bits to each level before coding is initiated. If we only account for the fact that level ℓ has four times as many coefficients as level $\ell + 1$ we get the "uniform" bit allocation

$$r_\ell(0) = \frac{4^{1-\ell}}{\sum_{j=1}^3 4^{1-j}} b' \quad (27)$$

Errors in low frequency information (higher ℓ) tend to be more damaging so that it is desirable to weight the bit allocations accordingly. In addition, the relative activity of coefficients at the different levels (e.g., variance in coefficient distributions) tends to be different at the different levels and change from array to array. The bit allocation given below extends (27) to include both of these factors. We let

$$r_\ell(0) \equiv \left(\frac{\beta_\ell a_\ell}{4^{\ell-1} A} \right) b' \quad (29)$$

where β_ℓ is an emphasis factor for level ℓ satisfying

$$\sum_{\ell=1}^3 \beta_\ell = 1. \quad (30)$$

$$a_\ell \equiv \frac{\mathcal{L}(\Phi[C'(\ell, s_\ell(0), 2)])}{(\text{no. of coefficients at level } \ell)} \quad (31)$$

is level activity, and

$$A \equiv \sum_{\ell=1}^3 \left(\frac{\beta_{\ell} a_{\ell}}{4^{\ell-1}} \right) \quad (32)$$

is array activity (see (6) - (10)).

Note that array activity, a_{ℓ} , is an estimate of the number of bits/coefficient required to code the coefficients of level ℓ under a fixed set of approximation conditions, namely $s_{\ell} = s_{\ell}(0)$ and $t_{\ell} = 2$. Typically $\beta_1 < \beta_2 < \beta_3$.

Selecting \bar{s}_{ℓ} and \bar{t}_{ℓ}

Once an initial allocation of bits has been established in (29) coding can begin, starting at level 3. The procedure is nearly the same at all levels, first $s_{\ell} = \bar{s}_{\ell}$ then $t_{\ell} = \bar{t}_{\ell}$ is determined such that $\hat{\mathcal{L}}(\Phi[C'(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})]) \approx r_{\ell}$ where r_{ℓ} is the allocated bits at level ℓ when coding begins. Recall that a control loop adjusts for errors in achieving the allocated bits at higher levels. Only at level 3 is it assured that $r_{\ell} = r_{\ell}(0)$.

Starting with $s_{\ell} = s_{\ell}(0)$, the algorithm for finding $s_{\ell} = \bar{s}_{\ell}$ changes s_{ℓ} until it is determined that either

$$\hat{\mathcal{L}}(\Phi[C'(\ell, s_{\ell}, t^*)]) \approx r_{\ell} (\pm 10\%) \quad (33)$$

or that s_{ℓ} is the largest such that

$$\hat{\mathcal{L}}(\Phi[C'(\ell, s_{\ell}, t^*)]) > r_{\ell} \quad (34)$$

where $t^* = 1$ if $s_{\ell} = 0$ and 2 otherwise. If (33) is satisfied for the chosen $s_{\ell} = \bar{s}_{\ell}$ then $\bar{t}_{\ell} = t^*$, otherwise \bar{t}_{ℓ} still needs to be determined. When (33) is not true for \bar{s}_{ℓ} , and $\ell = 2$ or 3, t_{ℓ} is incremented until either

$$\hat{\mathcal{L}}(\Phi[C'(\ell, \bar{s}_{\ell}, t_{\ell})]) \approx r_{\ell} \quad (35)$$

or that t_{ℓ} is the largest such that

$$\hat{\mathcal{L}}(\Phi[C'(\ell, \bar{s}_{\ell}, t_{\ell})]) > r_{\ell} \quad (36)$$

At level 1 this criterion is changed slightly to choosing the t_{ℓ} which results in the closest approximation to r_{ℓ} .

*Recall that $C(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})$ is obtained from $C'(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})$ by dividing all numbers by four. The symbol \wedge over a data set, such as $\hat{C}_0(\ell)$, means an approximation or estimate of the original numbers in $C_0(\ell)$.

Once \bar{s}_{ℓ} , \bar{t}_{ℓ} are determined the actual coding of $C'(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})$ takes place, the output appearing in the appropriate location in (24), preceded by a header identifying the choice of \bar{s}_{ℓ} and \bar{t}_{ℓ} . An implied control loop adjusts $r_{\ell-1}(0)$ to compensate for an accumulated difference between target rates and actual rates of higher level arrays.

INVERSE OPERATIONS

Direct Inverse*

Once the variable length coding in the sequence $\Phi[\hat{C}_0(0)]$ of (24) is unraveled, a straightforward inverse transform can be applied to the data sets $C_0(4)$, $C(4, 0, 1)$ and $C(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})$, $\ell = 1, 2, 3$ to retrieve an approximation to the original data array $C_0(0)$. The procedure is illustrated in Fig. 7.

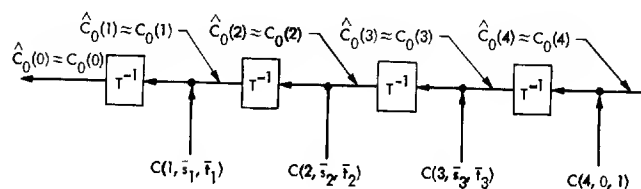


Fig. 7. Direct Inverse Transform.

Starting with level 4, each c_0 coefficient in $\hat{C}_0(\ell)$ (which is an approximation of a 2^{ℓ} by 2^{ℓ} array average) is combined with approximations to the corresponding c_1 , c_2 , c_3 coefficients in $C(\ell, \bar{s}_{\ell}, \bar{t}_{\ell})$ using the two by two inverse Hadamard transform in (13). Each operation produces approximations to four of the original c_0 terms from level $\ell - 1$, elements of $C_0(\ell - 1)$. Each term produced represents a $2^{\ell-1}$ by $2^{\ell-1}$ array average. The collective operations performed on all of $\hat{C}_0(\ell)$ is simply denoted by T^{-1} in Fig. 7 and yields the result $\hat{C}_0(\ell - 1)$. The procedure continues until $\ell = 0$ and the c_0 terms produced are approximations to the original array $C_0(0)$.

Adaptive Inverse

Simulations of images reconstructed through use of the direct inverse transform in Fig. 7 have

exhibited good to excellent rate/fidelity performance. However, such reconstructions are subject to the familiar, and subjectively annoying, squaring effects characteristic of many transform techniques. These effects can be substantially reduced, without loss in edge fidelity, by modifying the inverse operations to make use of the correlation between c_0 coefficients and the constraints imposed on c_1 , c_2 and c_3 coefficient errors by the operations $S_{\bar{s}_\ell}[\cdot]$ and $T_{\bar{t}_\ell}[\cdot]$.

Intermediate results. Let $\hat{\hat{C}}_0(\ell)$ be the result of applying the adaptive inverse algorithm at level $\ell + 1$, where $\hat{C}_0(\ell)$ denotes a refined estimate of the original c_0 coefficients at level ℓ , $C_0(\ell)$. Applying the direct inverse operations, T^{-1} , to $\hat{\hat{C}}_0(\ell)$ and $C(\ell, \bar{s}_\ell, \bar{t}_\ell)$ yields, as before, an approximation to the original c_0 coefficients in $C_0(\ell - 1)$. This time the set of numbers we obtain is an intermediate result we'll denote by $W_0(\ell - 1)$. Processing of $W_0(\ell - 1)$ results in $\hat{C}_0(\ell - 1)$ which can be used to proceed on to level $\ell - 2$. The procedure starts with $\hat{\hat{C}}_0(4) = C_0(4)$ and continues until $\hat{\hat{C}}_0(0)$ is obtained. Noting that the processing of $W_0(\ell - 1)$ is constrained by information in $C(\ell, \bar{s}_\ell, \bar{t}_\ell)$ and $\bar{s}_\ell, \bar{t}_\ell$, a block diagram of the procedure so far is given in Fig. 8. The symbol P_ℓ is used to denote the combined processing of all of $W_0(\ell - 1)$. P_ℓ is assumed to have no effect when $\bar{s}_\ell = 0$ and $\bar{t}_\ell = 1$ (e.g., when $\ell = 4$). γ_ℓ is a parameter to be specified later.

Natural spatial position. Before defining P_ℓ in detail note that the sets of numbers $C_0(\ell - 1)$, $\hat{C}_0(\ell - 1)$ and $W_0(\ell - 1)$ should be viewed as two dimensional arrays with each number located in its "natural spatial position." That is, the c_0 coefficient in $C_0(\ell - 1)$ which represents a particular $2^{\ell-1}$ by $2^{\ell-1}$ subarray average in

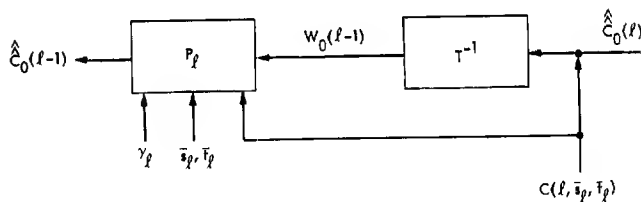


Fig. 8. Adaptive Inverse, Level ℓ .

$C_0(0)$ should occupy the corresponding spatial position in $C_0(\ell - 1)$. When $C_0(0)$ is a 64 by 64 data array, $C_0(\ell - 1)$, $\hat{C}_0(\ell - 1)$ and $W_0(\ell - 1)$ are $2^{6-(\ell-1)}$ by $2^{6-(\ell-1)}$ arrays. Since the numbers in $W_0(\ell - 1)$ and $C_0(\ell - 1)$ represent $2^{\ell-1}$ by $2^{\ell-1}$ averages, when $\ell - 1 > 0$ the elements of both can be expected to have two dimensional correlation properties similar to those of sampled data arrays, provided the natural ordering is maintained. As a consequence, simple estimation procedures which are effective when applied to data arrays $C_0(0)$ and $W_0(0)$ can be expected to be effective when applied to $C_0(\ell - 1)$ and $W_0(\ell - 1)$ for $\ell - 1 > 0$.

Two by two processing. The algorithm for changing the intermediate $W_0(\ell - 1)$ into $\hat{C}_0(\ell - 1)$ operates on individual two by two arrays in a naturally ordered $W_0(\ell - 1)$, one at a time and in a specified order (e.g., first row, second row, etc.). The processed numbers from a given two by two may be used in processing subsequent two by twos. When all two by twos have been processed the combined result is $\hat{C}_0(\ell - 1)$. Except for the number of two by twos processed, the algorithm performed at each level is identical.

Let

$$\tilde{W} \equiv \begin{bmatrix} w_0 & w_1 \\ w_3 & w_2 \end{bmatrix} \quad (37)$$

be an unprocessed two by two array of $W_0(\ell - 1)$ which is not on the border of $W_0(\ell - 1)$. Also let $E_{i,j}$ be elements of $W_0(\ell - 1)$, processed or not, which spatially surround \tilde{W} as in Fig. 9.

With $\gamma_{\ell-1}$ a level $\ell - 1$ parameter satisfying $0 \leq \gamma_{\ell-1} \leq 1$, each w_i is re-estimated using surrounding data as

$$\hat{w}_i \equiv \gamma_{\ell-1} w_i + \frac{(1 - \gamma_{\ell-1})}{3} \sum_j E_{i,j} \quad (38)$$

giving the array

$$\hat{\tilde{W}} \equiv \begin{bmatrix} \hat{w}_0 & \hat{w}_1 \\ \hat{w}_3 & \hat{w}_2 \end{bmatrix} \quad (39)$$

Note that errors in this estimate due to errors in the $\{E_{i,j}\}$ tend to be compensated

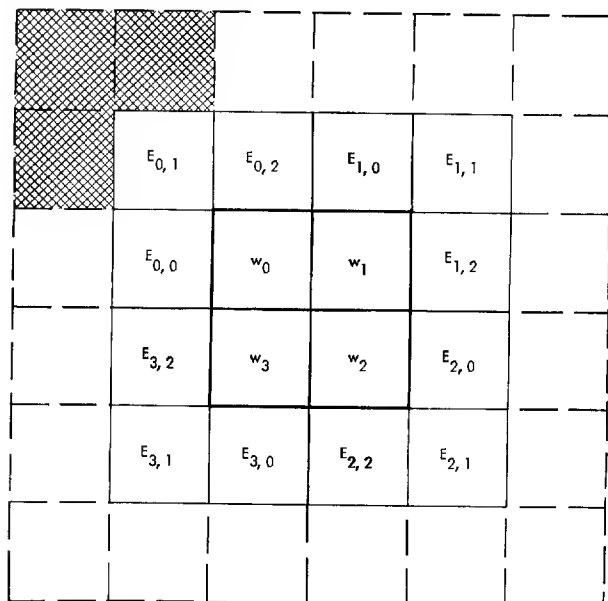


Fig. 9. Two by Two and Surrounding Data.

for by the fact that, for a given i , each $E_{i,j}$ comes from a different two by two.

If we apply the two by two Hadamard transform to both \tilde{W} and \hat{W} using (14) we obtain the sets of coefficients

$$\begin{aligned} x_k(\bar{s}_\ell, \bar{t}_\ell) &\equiv \langle \tilde{W}, \tilde{H}_k \rangle \\ &\text{and} \\ \hat{x}_k(0, 1) &\equiv \langle \hat{W}, \hat{H}_k \rangle \end{aligned} \quad (40)$$

Now $x_1(\bar{s}_\ell, \bar{t}_\ell)$, $x_2(\bar{s}_\ell, \bar{t}_\ell)$ and $x_3(\bar{s}_\ell, \bar{t}_\ell)$ are transform coefficients as transmitted, that is, elements of $C(\ell, \bar{s}_\ell, \bar{t}_\ell)$. Whereas $\hat{x}_1(0, 1)$, $\hat{x}_2(0, 1)$ and $\hat{x}_3(0, 1)$ represent estimates, from surrounding data, of the original coefficients in $C(\ell, 0, 1)$ before shifting and thresholding. If the $\{\hat{x}_k(0, 1)\}$ are indeed valid estimates of the true coefficients, then operating on them with $S_{\bar{s}_\ell}[\cdot]$ and $T_{\bar{t}_\ell}[\cdot]$ should produce the same set of transmitted numbers $\{x_k(\bar{s}_\ell, \bar{t}_\ell)\}$. We next make use of this principal to guide a decision to use $x_k(\bar{s}_\ell, \bar{t}_\ell)$, $\hat{x}_k(0, 1)$ or some modification of $\hat{x}_k(0, 1)$.

In order to make more direct use of equations developed earlier we let

$$x'_k(\bar{s}_\ell, \bar{t}_\ell) \equiv 4x_k(\bar{s}_\ell, \bar{t}_\ell) \text{ and } \hat{x}'_k(0, 1) \equiv 4\hat{x}_k(0, 1) \quad (42)$$

where we recall that $x'_k(\bar{s}_\ell, \bar{t}_\ell)$ is simply the stored integer form of $x_k(\bar{s}_\ell, \bar{t}_\ell)$ (see (15), (17) and Fig. 6).

Making use of (16) and (20), the final form of any estimate $x'_k(0, 1)$ must satisfy

$$x'_k(\bar{s}_\ell, \bar{t}_\ell) = 2^{\bar{s}_\ell} T_{\bar{t}_\ell} \left[2^{-\bar{s}_\ell} S_{\bar{s}_\ell} [\hat{x}'_k(0, 1)] \right] \quad (42)$$

Here we will treat only the case of primary interest where $x'_1(\bar{s}_\ell, \bar{t}_\ell) = x'_2(\bar{s}_\ell, \bar{t}_\ell) = x'_3(\bar{s}_\ell, \bar{t}_\ell) = 0$.

We define

$$\begin{aligned} \Lambda &\equiv \max_k \left| 2^{-\bar{s}_\ell} S_{\bar{s}_\ell} [\hat{x}'_k(0, 1)] \right|, \\ \alpha &\equiv \frac{\bar{t}_\ell - 0.5}{\Lambda} \end{aligned} \quad (43)$$

Then if $\Lambda < \bar{t}_\ell$ we choose the final form for the two by two array \tilde{W} to be \hat{W} . If $\Lambda \geq \bar{t}_\ell$ we obtain this output by applying the inverse two by two Hadamard transform to

$$\begin{bmatrix} x_0(\bar{s}_\ell, \bar{t}_\ell) & \alpha \hat{x}_1(0, 1) \\ \alpha \hat{x}_3(0, 1) & \alpha \hat{x}_2(0, 1) \end{bmatrix} \quad (44)$$

We recognize α as a scaling factor which makes $\alpha \hat{x}_k(0, 1)$ satisfy (42).

The condition in (42) is much more restrictive when some of the transmitted coefficients are non-zero. As a result the algorithm presently simulated will tend to accept \tilde{W} unmodified a large percentage of the time. It basically leaves edges alone.

Observe that when \tilde{W} lies on the border of $W_0(\ell - 1)$ the surrounding data, $E_{i,j}$, may still be available. Provided the particular \tilde{W} does not lie on the border of a picture, the necessary $E_{i,j}$ can be obtained from the corresponding transform level of adjacent 64 by 64 data arrays. Present simulations have shown that this eliminates noticeable transitions between adjacent data arrays even at very low data rates.

Comments. The adaptive inverse algorithm described above has not been shown to be optimum in any fashion. It is however both practical and effective in improving subjective fidelity and in reducing mean-square-error. It is felt that both simulations and analysis could lead to more sophisticated adaptive versions with worthwhile additional performance benefits.

RM2 IMPLEMENTATION

Based upon preliminary implementation studies, it was estimated that 300 to 400 CMOS integrated circuits would be required to implement the RM2 compression algorithm in the form described in this report. Such a design would accommodate an 80 KBS input data rate. More restrictive applications could reduce this substantially although the 300-400 chips is considered quite practical. Continuing advances in solid state technology promise to further minimize the chip count.

It is of interest to note that the transform implementation is almost trivial since it requires no more than an iterative application of a two by two fast Hadamard transform.^[6] The rate allocation algorithms could ideally be performed by microcomputers, a likely element in future flight data systems. This was not assumed in the estimates however. Much higher speeds could be accommodated by using a more parallel structure than the one assumed.

SIMULATIONS

A study was recently initiated to investigate the applicability of AICS to advanced Pioneer missions which incorporate imaging. Sample images have been selected for this purpose by imaging scientists to be representative of expected mission targets, and via processing, to exhibit characteristics similar to those of a planned Pioneer CCD camera. We will present some preliminary pictorial results.

Mercury Closeup

One of the sample images to be used in this study is a 512 by 768 closeup of Mercury derived from the recent Mariner flybys. A standard stretched display of this scene, including

histograms, is given in Fig. 10. A more revealing display is shown in Fig. 11. Here a two dimensional filter which tends to equalize average brightness throughout the scene was applied. This permits a more severe contrast stretch to be used, enhancing visual detail on film. These operations are representative of the block labeled "image processing" in the overall AICS system diagram in Fig. 2.

RM2 Compressed

Examples of images compressed by RM2 at rates of 2.0, 1.33, 1.0, 0.66 and 0.5 bits/pixel (compression factors of 4, 6, 8, 12 and 16) are shown in Figs. 12-16 respectively. The same critical display technique of Fig. 11 is used. Except for the rate; all internal parameters of RM2 were identical in each case. These are noted in Table 1. The reader may refer to the appropriate section for further details. In addition, a simple black space detection algorithm was employed (see (4) and (5)) to illustrate the idea.

Table 1. Parameters for Simulation Examples

Parameters	Used for	Where
Picture Loop Gain = 1.0	Rate Control Within Imaging Sequence	Fig. 3
Array Loop Gain = 0.35	Rate Control Within an Image	Fig. 4
$e(m)$, $m = 1, 2, \dots, 8$ 0.0, 0.5, 1.0, 1.5, 1.9, 2.2, 2.5, 2.8	Rate Allocation to picture arrays	Equa. 6, 7
$n' \alpha_{\text{MIN}} = 0.35$	Rate Allocation to picture arrays	Equa. 6-10
Initial shifts $s_1(0) = s_2(0)$ $= 2$, $s_3(0) = 1$	Obtaining Array Activity; starting point for finding \bar{s}_l .	Equa. 31-34
$\beta_1 = 0.2$, $\beta_2 = 0.3$, $\beta_3 = 0.5$	Frequency Emphasis	Equa. 29-32
$\gamma_1 = \gamma_2 = \gamma_3$ $= 0.6$	Adaptive Inverse Estimation Parameters	Equa. 38

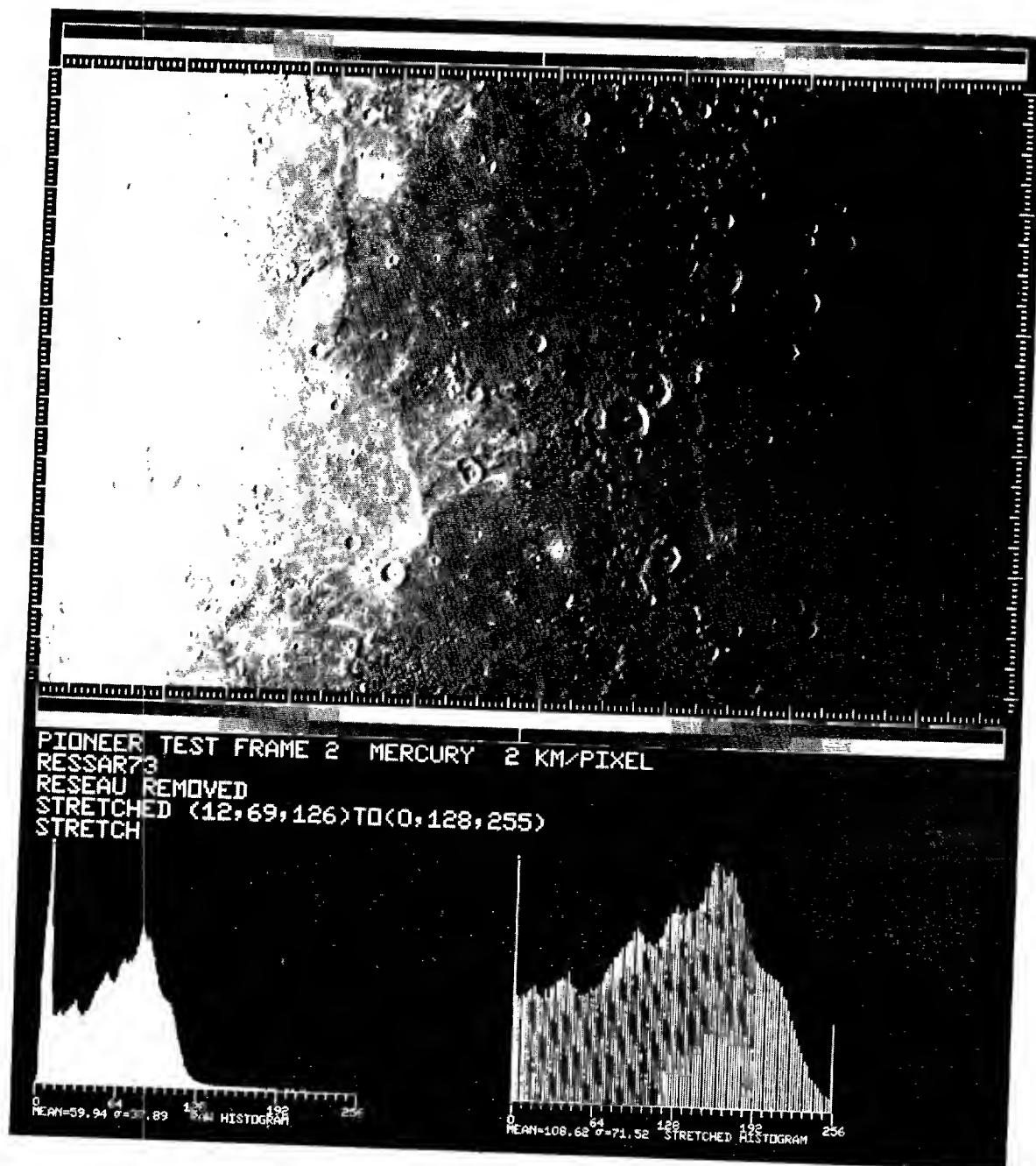


Fig. 10. Stretched Pioneer Test Frame.

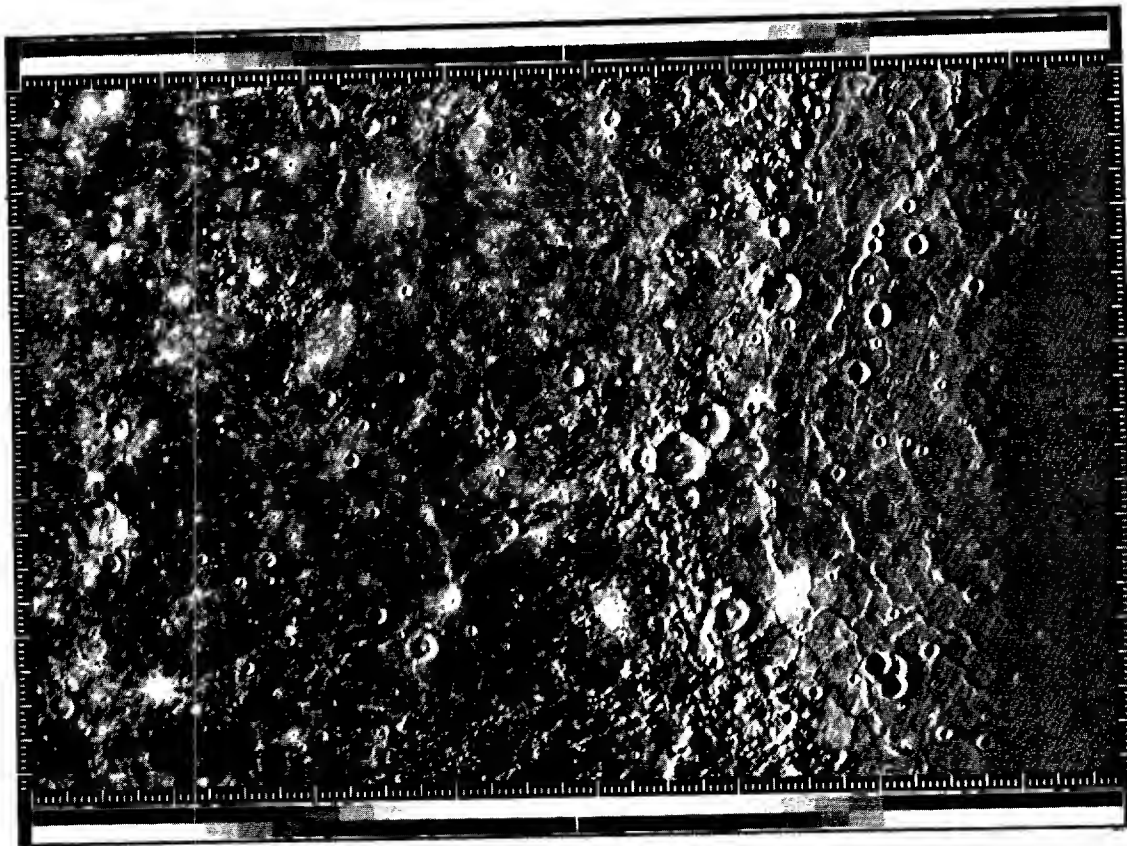


Fig. 11. Mercury Closeup Original, Filtered.

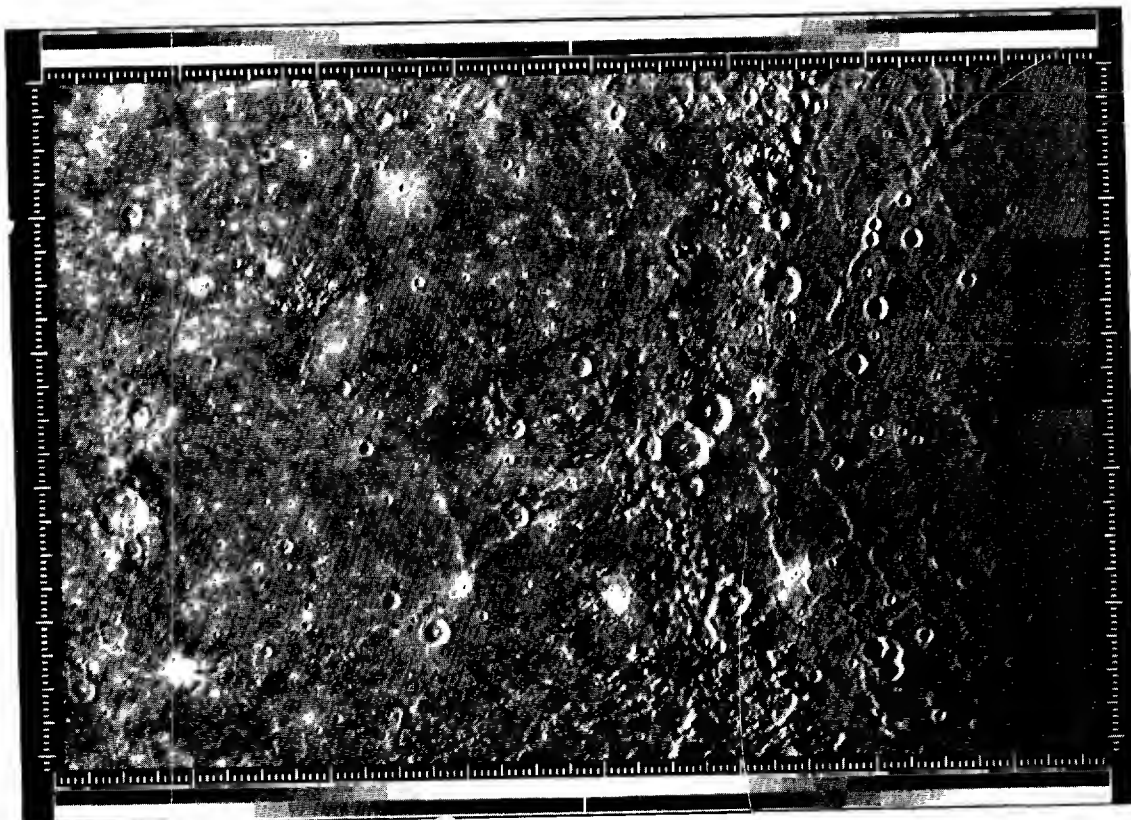
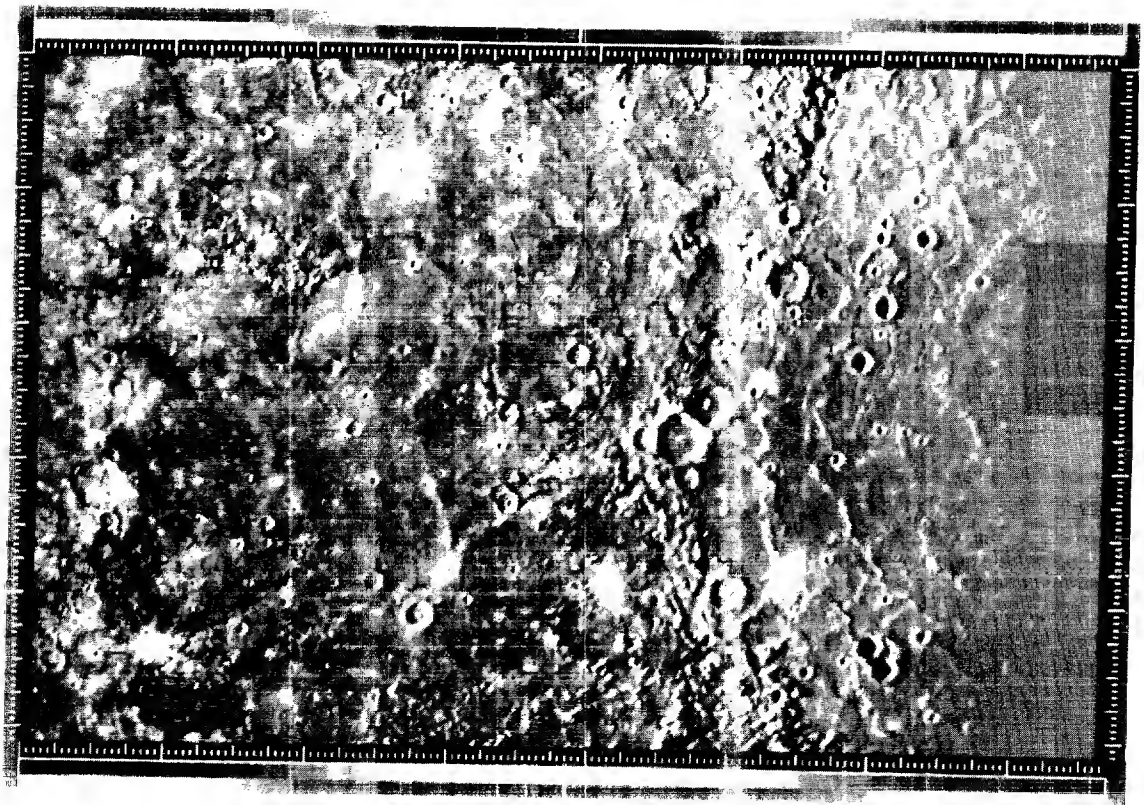
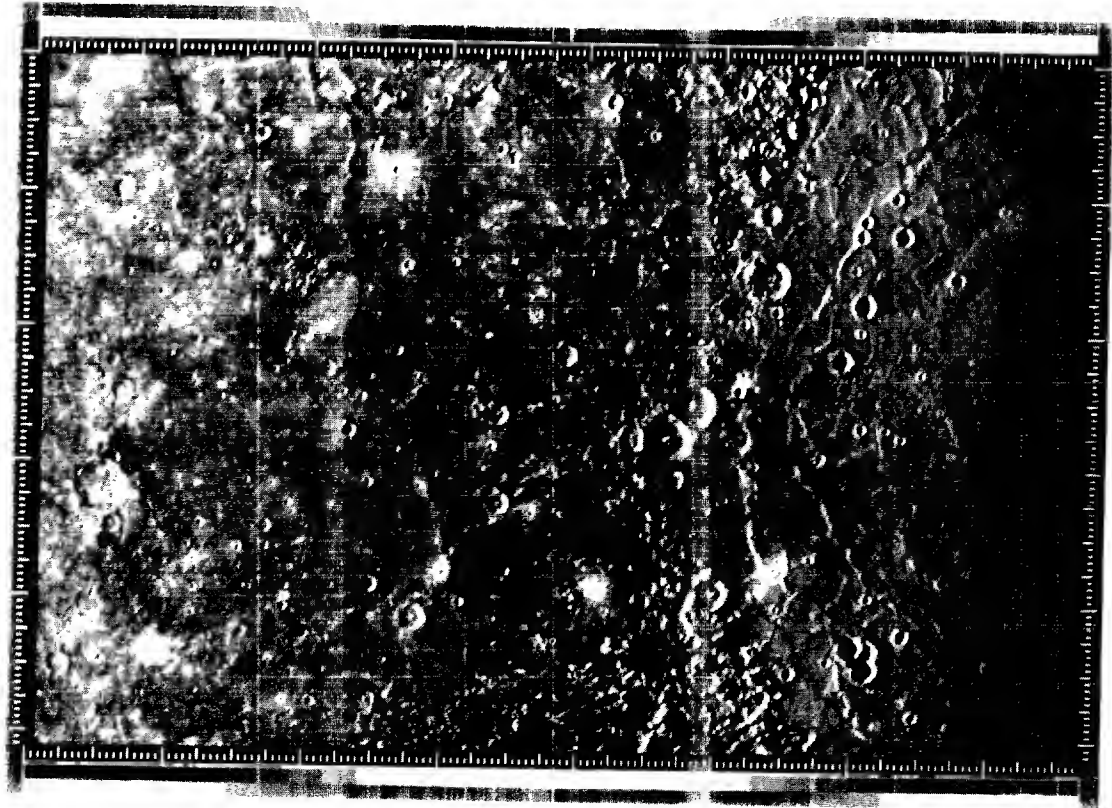


Fig. 12. RM2 Coded to 2.0 bits/pixel (4:1)



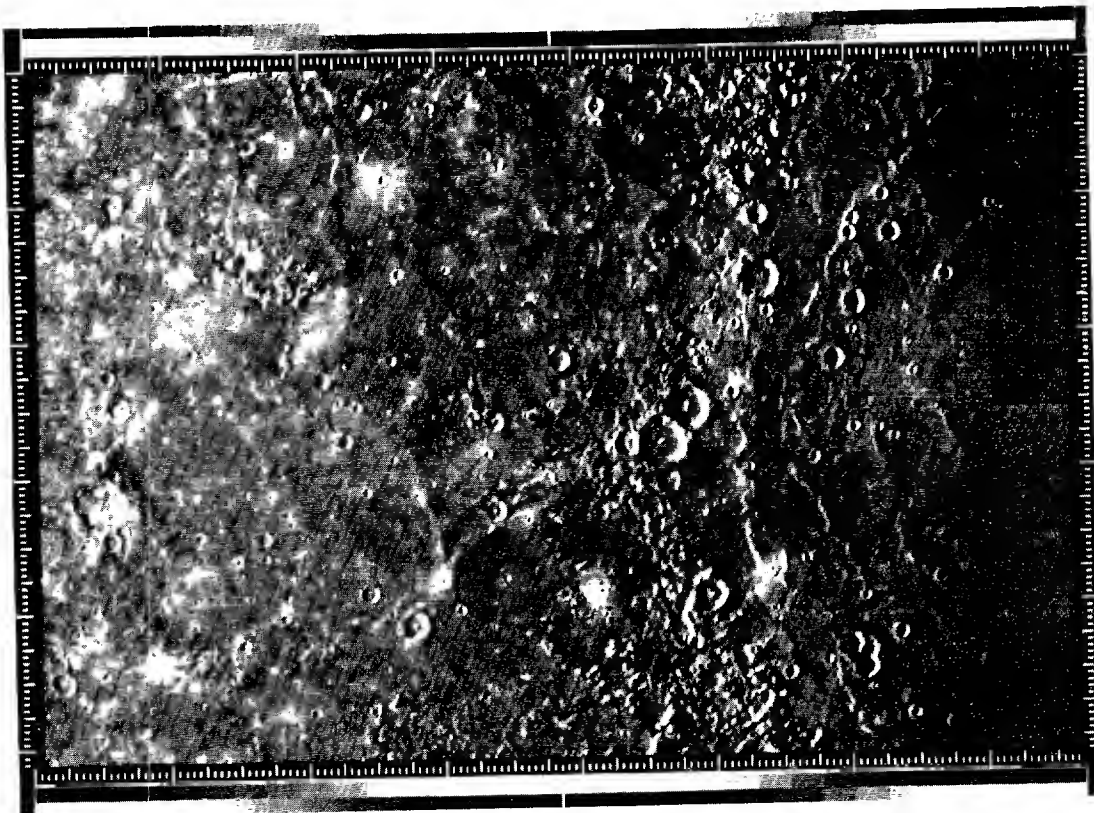


Fig. 15. RM2 Coded to 0.66 bits/pixel (12:1)

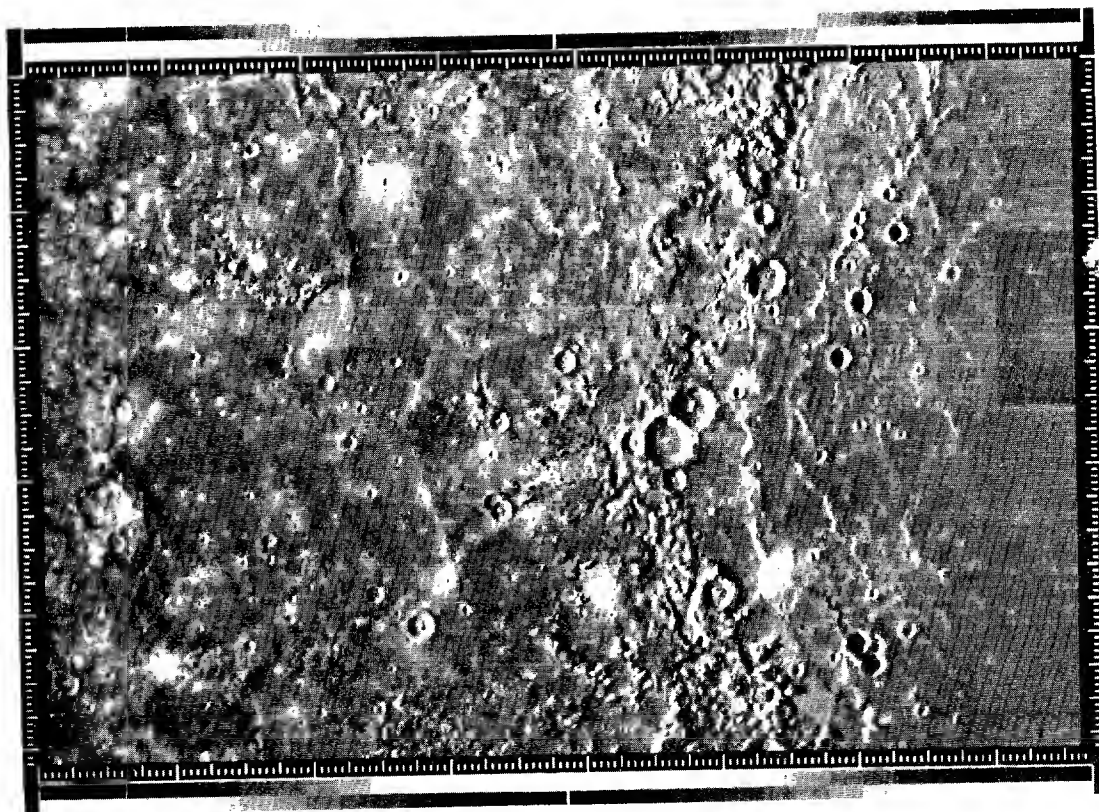


Fig. 16. RM2 Coded to 0.5 bits/pixel (16:1)

Rate distribution. The top and bottom of the Mercury picture were separately treated as a sequence of two 256 by 768 images. It is of interest to look at the initial rate allocation to each of the 48 arrays making up the top half as a result of the algorithm described in (5) - (9). This demonstrates one aspect of the adaptive nature of RM2. The distribution of array activities derived from (32) is shown in Fig. 17.

Choosing the image rate of 1.33 bits/pixel as an example (Fig. 13), the application of the algorithm in (5)-(9) yields the initial rate allocations in bits/pixel to each of these arrays shown in Fig. 18.

Observe that the two arrays in the lower righthand corner have been allocated a rate of zero bits/pixel. These have actually been edited (see (5)). No real data has been lost as can be noted from Figs. 10 and 11. Without the editing they would have been assigned a rate of around 0.5 bits/pixel, the same as for the low activity arrays immediately above. The rate assignments to other areas reflect their relative activity, with one extremely active array receiving over 3 bits/pixel.

Control loop error. At the completion of the top half picture the rate error (ϵ_i in Fig. 3) was 1125 bits or 0.006 bits/pixel. All these "unused bits" were used to adjust the rate of the bottom half picture (i.e., a picture loop gain of 1). At all other compression rates $|\epsilon_i|$ was quite similar.

Rate/fidelity. Recall that because of the concatenated Reed-Solomon/Viterbi Channel in Fig. 2 any rate/fidelity tradeoffs provided by RM2, as in the above sequence of images, represent true end-to-end communication system tradeoffs. Further, the user could specify any intermediate

rate and expect a smooth monotonic relationship between the rate specified and the fidelity obtained. These considerations and a preliminary assessment of RM2 performance from these examples and other simulation results gives AICS in Fig. 2 an effective average information rate capability of 3 to 5 times that of the Jupiter/Saturn Communication System in Fig. 1. Similar comparisons with an uncoded communication system, such as that of the recent Mariner Venus/Mercury flybys, would yield an AICS information rate advantage of up to 10 dB.

A more extensive evaluation of RM2 when applied to planetary imaging data will come from the Pioneer study. Applications of RM2 to other problems is being investigated and initial results have been excellent. In addition to tailoring the simulation version of RM2 described here to specific applications, further improvements in performance can be expected from extensions and modifications to the algorithm. Particular emphasis here should be placed on expanding the rate allocation concepts and improving the adaptive inverse. Also important is the specification of simpler versions of RM2 for more restrictive applications in which being extremely adaptive is not important.

REFERENCES

1. Rice, R. F., "Channel coding and data compression system considerations for efficient communication of planetary imaging data," Technical Memorandum 33-695, Jet Propulsion Laboratory, Pasadena, California, June 15, 1974.
2. J. P. Odenwalder et al., "Hybrid coding systems study," submitted to NASA Ames Res. Ctr. by Linkabit Corp., San Diego, Calif., Final Rep., Contract NAS2-6722, Sept. 1972.

0.47	0.43	0.54	0.55	0.50	0.42	0.55	0.69	0.95	0.47	0.31	0.03
0.49	0.53	0.49	0.34	0.28	0.33	0.43	0.63	0.63	0.44	0.25	0.07
0.42	0.46	0.60	0.44	0.37	0.41	0.38	0.66	0.62	0.47	0.25	0.02
0.46	0.40	0.37	0.43	0.43	0.51	0.61	0.81	0.40	0.42	0.22	0.02

Fig. 17. Array (64 by 64) Activities for Top Half Mercury Closeup.

1.38	1.29	1.57	1.61	1.48	1.25	1.60	2.06	3.11	1.38	1.00	0.50
1.43	1.55	1.43	1.06	0.931	1.05	1.27	1.88	1.87	1.30	0.88	0.56
1.24	1.36	1.75	1.31	1.13	1.22	1.16	1.95	1.84	1.39	0.86	0.00
1.36	1.22	1.13	1.29	1.29	1.50	1.79	2.52	1.19	1.25	0.82	0.00

Fig. 18. Initial Rate Allocations for Average Rate of 1.33 bits/pixel.

3. J. P. Odenwalder, "Concatenated Reed-Solomon/Viterbi channel coding for advanced planetary missions: analysis, simulations and tests," submitted to Jet Propulsion Laboratory by Linkabit Corp., San Diego, Calif., Final Rep., Contract 953866, Dec. 1, 1974.
4. J. A. Heller and I. M. Jacobs, "Viterbi decoding for satellite and space communication," IEEE Trans. Commun. Technol., Vol. COM-19, part II, Oct. 1971, pp. 835-848.
5. "Deep Space Network/Flight Project Interface Design Handbook," JPL Document 810-5, Rev. C, Jet Propulsion Laboratory, Pasadena, Calif., April 15, 1972 (JPL Internal Document).
6. R. F. Rice, "RM2: transform operations," Technical Memorandum 33-680, Jet Propulsion Laboratory, Pasadena, California, March 1, 1974.
7. R. F. Rice and J. R. Plaunt, "Adaptive variable length coding for efficient compression of spacecraft television data," IEEE Trans. Commun. Technol., Vol. COM-19, part I, Dec. 1971, pp. 889-897.

ADVANCED FACSIMILE SYSTEM AND NETWORK APPLICATIONS

John W. Watkins
Harris Corporation, Electronic Systems Division
Melbourne, Florida 32901

Introduction

Laser, acousto-optics and dry silver material technologies were designed into an advanced facsimile system to achieve high recording speeds, high copy resolution, and chemical-free operation. The basic system was designed so that its performance specifications can be modified over a wide range with minor changes required to enable a broad spectrum of applications to be addressed.

The technologies that were used and the design features of the facsimile system are reviewed in this report, including a description of the unique optics bench that was designed to provide electronic control of its performance parameters. These unique features, along with the modular design of the system, provide the versatility required to satisfy a wide range of network applications.

Laserfax Design Objectives

Facsimile and hard copy display applications have been found to require a relatively wide range of operational and functional specifications. While it would be difficult and expensive, if not impossible, to develop a single system that would satisfy all requirements, an approach has been devised by which a significant segment of the applications can be addressed through a functionally designed system whose specifications can be modified significantly without requiring a major redesign.

It was first determined that although the facsimile system to be developed would be capable of transmitting and reproducing printed documents, its principal capability would be to satisfy those applications that require quality photographic reproductions. Therefore, the system trade-off for a specific application is usually between the desired spatial resolution in the output copy and the transmission time over a predefined data link. The dynamic range or the number of gray shades in the reproduced photograph is normally not a significant trade-off parameter. Thus, the next constraint that was placed upon the design was to require that the resolution of the system be easily modifiable with the document scanning rate being treated as the dependent variable.

With new telephone, microwave and CATV data links becoming more available, it was also desirable for the overall speed of the system to be changeable while causing a minimum impact to the basic system design.

In addition to the system versatility described above, the equipment would have many convenient operational features such as, copy reproduction on a dry recording material, low cost per copy, daylight loading of the recording material, ease of operation with no operator required at the receiver, low maintenance, high reliability, easy access to all sub-assemblies for service, and others.

System Configuration and Operation

The drawings presented in Figures 1 and 2 show cross-sectional outlines of the major subassemblies in the facsimile transmitter and receiver, respectively. In the transmitter, the document or photograph to be transmitted is placed in the input chute. Upon command, the capstan drive translates the document past the scanning laser beam that is projected from the optics bench and that is brought to focus on the document as depicted in the figure. With the intensity of the laser beam being held constant from the optics bench, the light that is scattered by the document and that falls onto the detector is therefore proportional to the reflectivity of the photograph or document being scanned. This video signal is then amplified and conditioned for transmission over the data link that has been selected for the particular application.

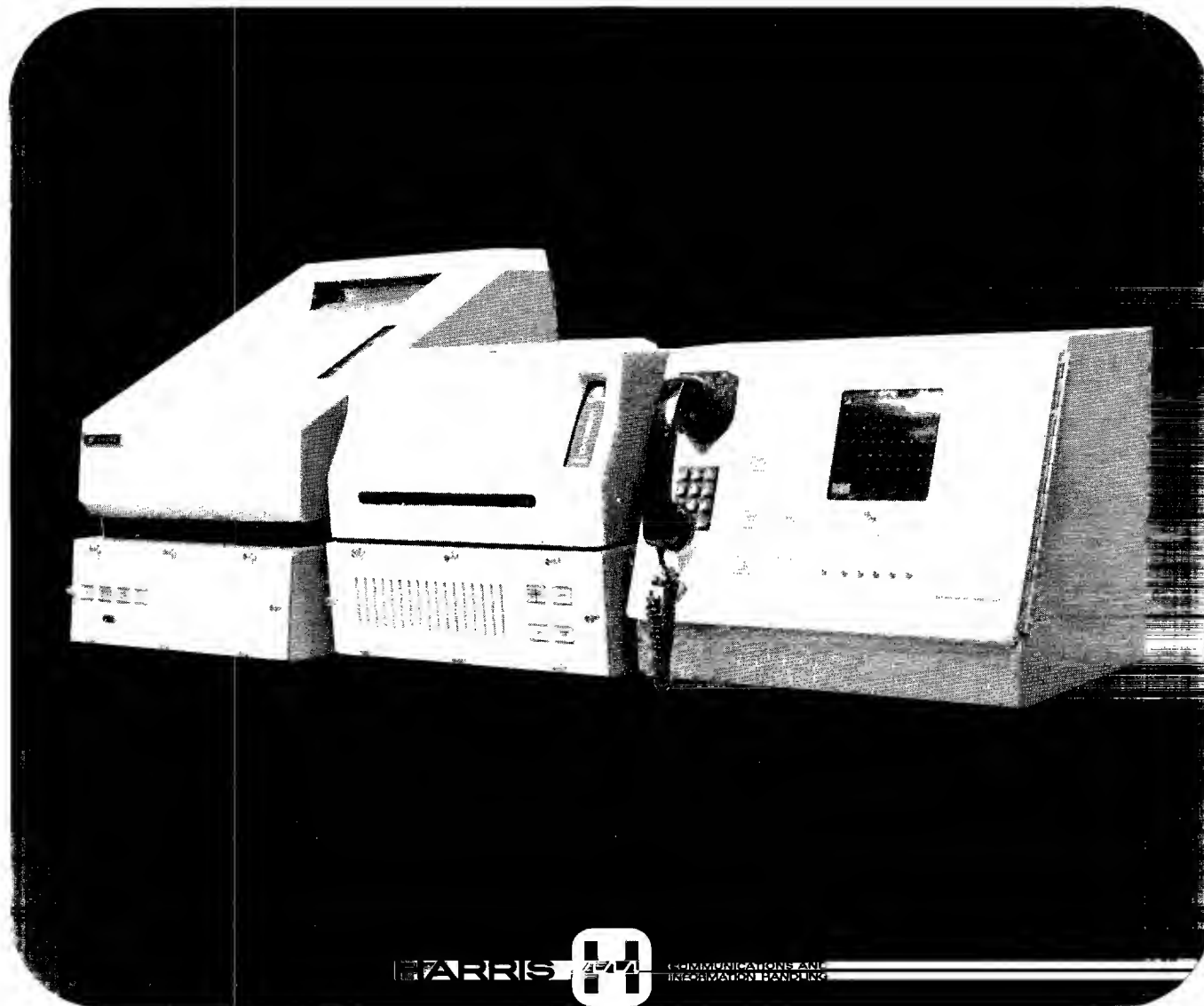
In the receiver, unexposed dry silver paper (3M Type 7771) is fed from the supply cassette around the capstan drive and past the exposure station where the modulated laser beam from the optics bench is brought to focus. After exposure, the paper progresses up through the paper feed mechanism towards the heat processor. When the transmission is complete, the paper is brought forward until the exposed paper passes the cutter bar where it is then cut. The unexposed paper is then retracted to the exposure station and the exposed sheet progresses through the heat processor where it is developed and delivered out the exit chute into the stacking tray.

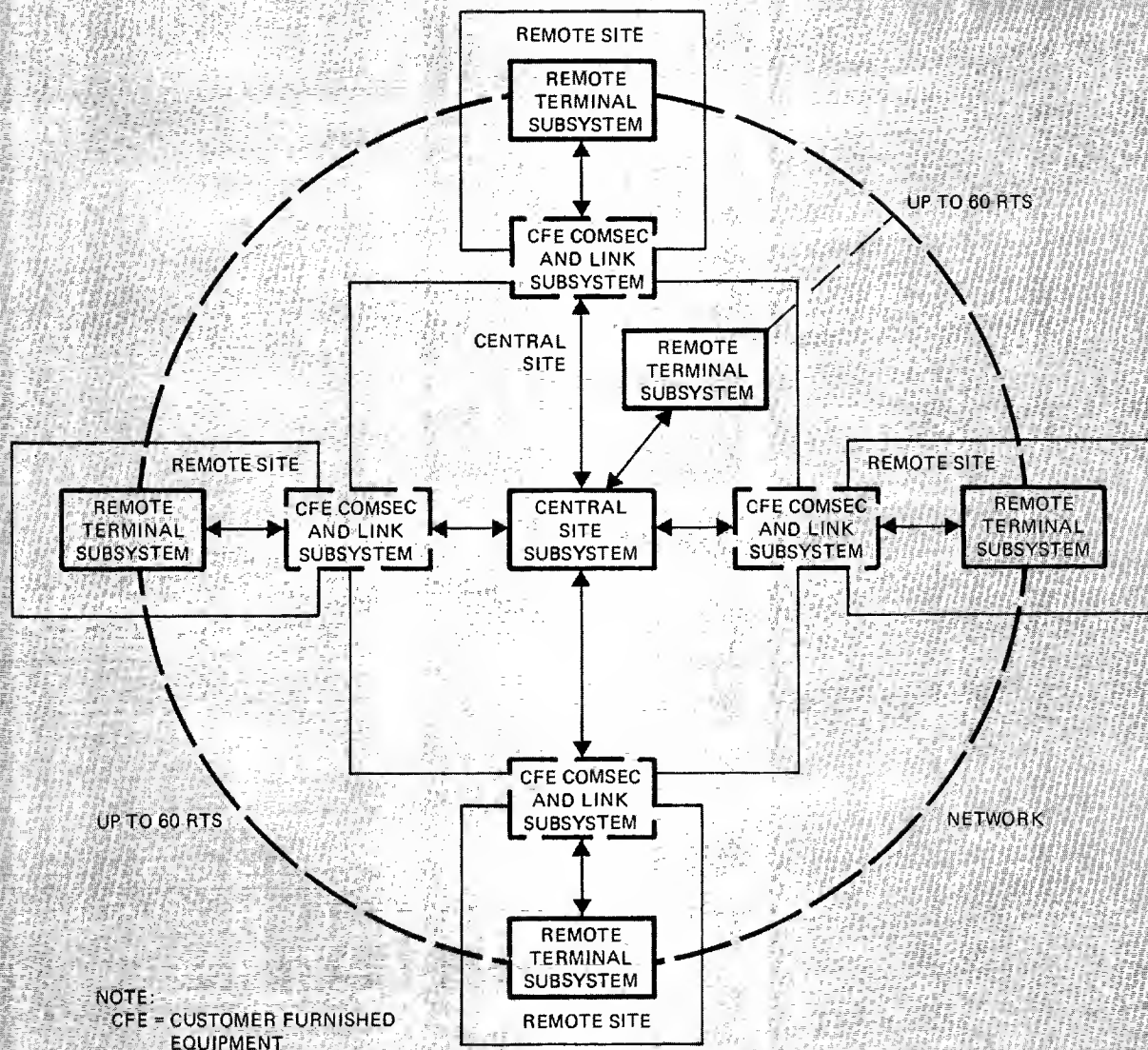
The major subassemblies in the receiver are the capstan drive, the paper supply cassette, the heat processor, the optics bench and the electronics assembly; and in the transmitter, they are the capstan drive, the optics bench and the electronics

Imagery Transmission System (ITS)

The Harris ITS is a high speed, photographic quality, secure, digital facsimile, switched network which is currently installed and can provide full duplex connections for up to 60 subscribers.

- | | |
|---|--|
| <input type="checkbox"/> Secure | <input type="checkbox"/> Automatic Unattended Receiver Operation |
| <input type="checkbox"/> Digital T-1 Rates (1.544 Mb/s) | <input type="checkbox"/> Automatic Opaque/Transparency Scan |
| <input type="checkbox"/> One Minute Transmission | <input type="checkbox"/> 285 Scan Lines Per Inch, 6-Bit Quantization |
| <input type="checkbox"/> Full Duplex, Simultaneous Voice/Data | <input type="checkbox"/> Dry Silver Output |
| <input type="checkbox"/> TEMPEST Qualified | <input type="checkbox"/> Switch-Selectable Contrast Enhancement |
| <input type="checkbox"/> Multi-Terminal Broadcast Connections | <input type="checkbox"/> Loopback Fault Diagnosis |





The central site consists of a computer-controlled nonblocking space-division solid-state switch capable of providing point-to-point and broadcast connections on prearranged subscriber lists. The ITS terminal consists of a Harris facsimile transmitter/receiver pair and a voice-control unit which are connected in a network arrangement to provide full-duplex simultaneous voice and imagery data over secure T-1 digital lines. The terminal operates at 285 scan lines per inch and 6-bit pixel quantization to provide extremely high quality photographic images on 3M Dry Silver paper. Transmission time is one minute over 1.544 Mb/s (T-1) data links conditioned through standard COMSEC interface

equipment. Various terminal modes are available including unattended receiver operation. A status panel provides the operator with continuous status display of all terminals on the network including in-service, out-of-service and busy conditions. The facsimile transmitter is capable of automatically detecting and scanning either opaque paper copies or transparent film copies. Up to four object enhancement curves with PROM density transfer functions are provided to modify the imagery for shadow stretch or glint enhancement. Several loop-back modes are designed to ease maintenance by quickly isolating system and terminal faults.

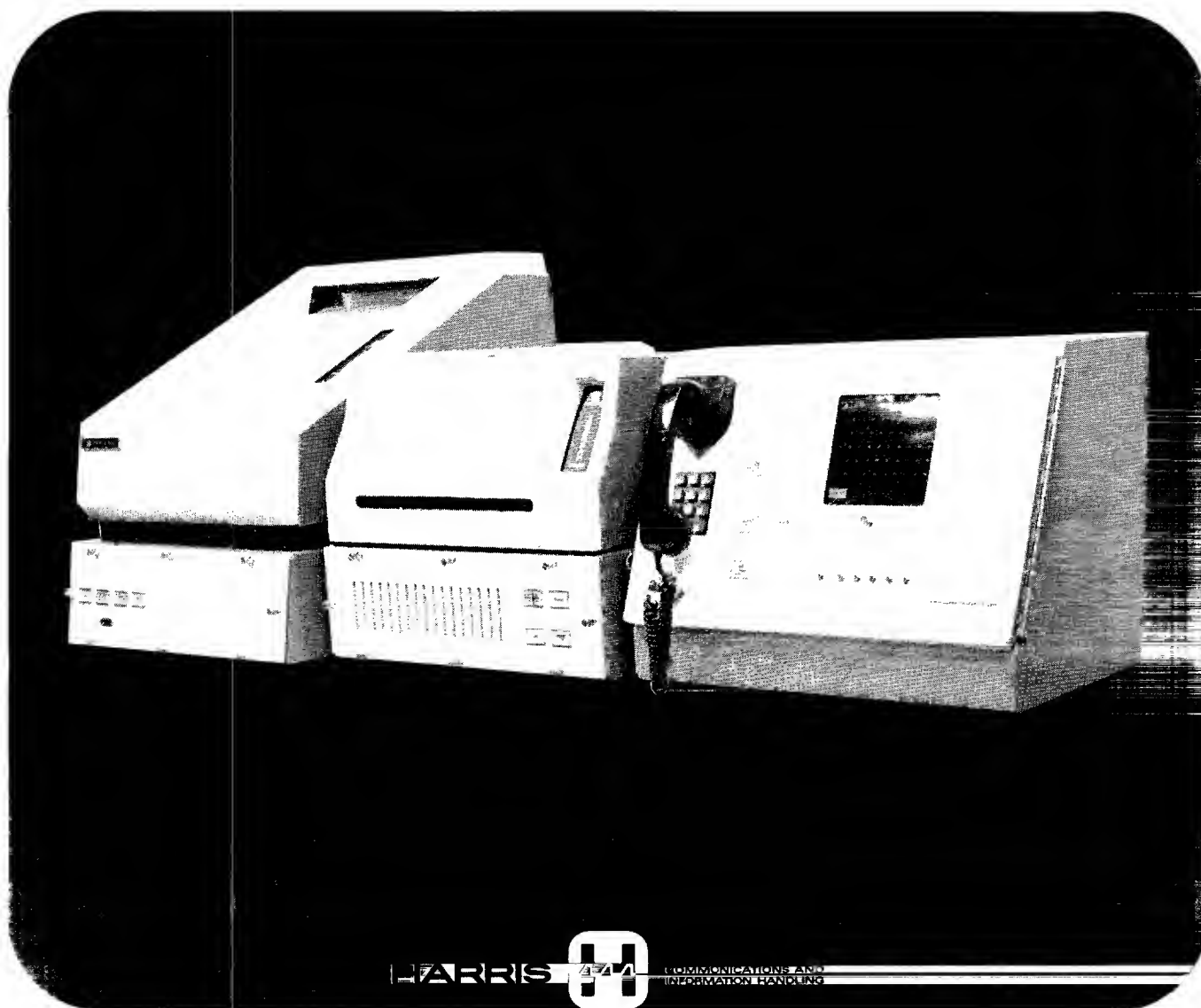


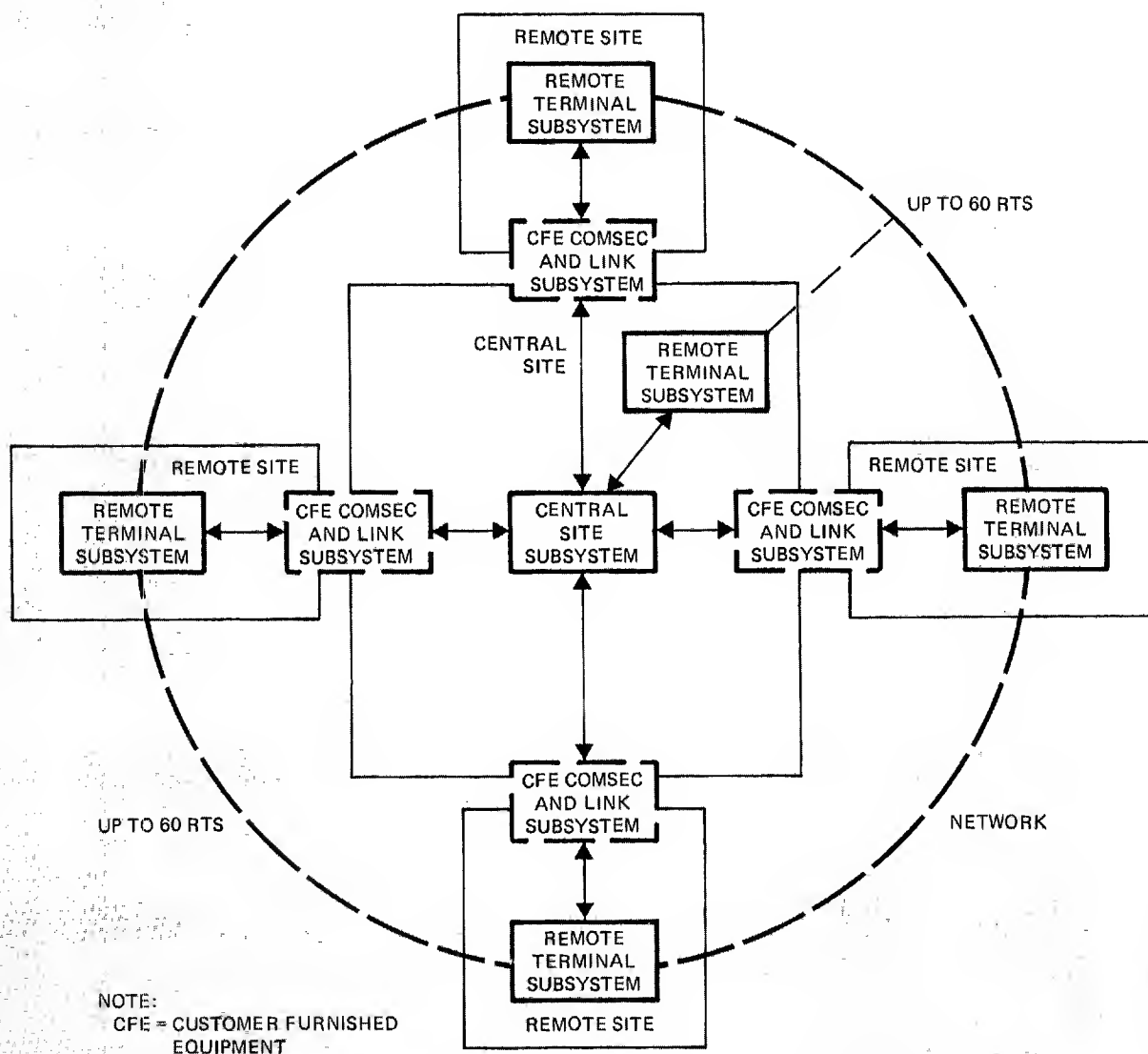
HARRIS CORPORATION Government Systems Group
Communications Systems P.O. Box 37, Melbourne, FL 32901 (305) 727-4311

Imagery Transmission System (ITS)

The Harris ITS is a high speed, photographic quality, secure, digital facsimile, switched network which is currently installed and can provide full duplex connections for up to 60 subscribers.

- | | |
|---|--|
| <input type="checkbox"/> Secure | <input type="checkbox"/> Automatic Unattended Receiver Operation |
| <input type="checkbox"/> Digital T-1 Rates (1.544 Mb/s) | <input type="checkbox"/> Automatic Opaque/Transparency Scan |
| <input type="checkbox"/> One Minute Transmission | <input type="checkbox"/> 285 Scan Lines Per Inch, 6-Bit Quantization |
| <input type="checkbox"/> Full Duplex, Simultaneous Voice/Data | <input type="checkbox"/> Dry Silver Output |
| <input type="checkbox"/> TEMPEST Qualified | <input type="checkbox"/> Switch-Selectable Contrast Enhancement |
| <input type="checkbox"/> Multi-Terminal Broadcast Connections | <input type="checkbox"/> Loopback Fault Diagnosis |





The central site consists of a computer-controlled nonblocking space-division solid-state switch capable of providing point-to-point and broadcast connections on prearranged subscriber lists. The ITS terminal consists of a Harris facsimile transmitter/receiver pair and a voice-control unit which are connected in a network arrangement to provide full-duplex simultaneous voice and imagery data over secure T-1 digital lines. The terminal operates at 285 scan lines per inch and 6-bit pixel quantization to provide extremely high quality photographic images on 3M Dry Silver paper. Transmission time is one minute over 1.544 Mb/s (T-1) data links conditioned through standard COMSEC interface

equipment. Various terminal modes are available including unattended receiver operation. A status panel provides the operator with continuous status display of all terminals on the network including in-service, out-of-service and busy conditions. The facsimile transmitter is capable of automatically detecting and scanning either opaque paper copies or transparent film copies. Up to four object enhancement curves with PROM density transfer functions are provided to modify the imagery for shadow stretch or glint enhancement. Several loop-back modes are designed to ease maintenance by quickly isolating system and terminal faults.



HARRIS

HARRIS CORPORATION Government Systems Group
Communications Systems P.O. Box 37, Melbourne, FL 32901 (305) 727-4311

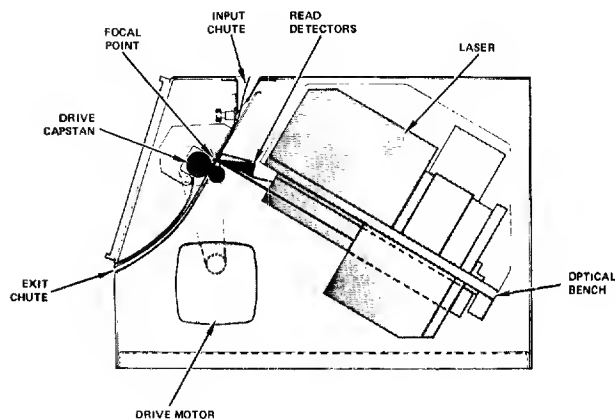


Fig. 1. Facsimile transmitter system configuration.

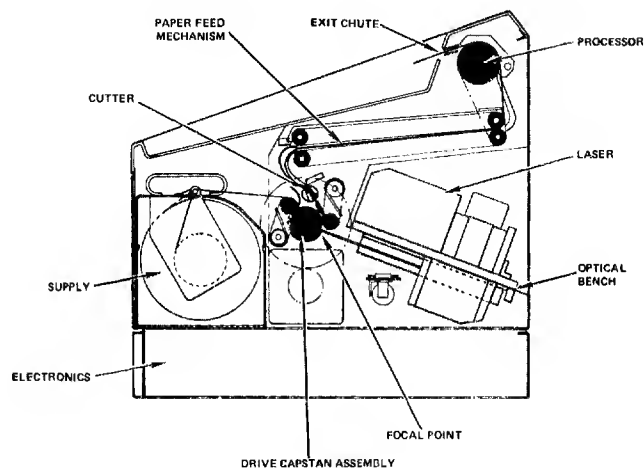


Fig. 2. Facsimile receiver system configuration.

assembly. Each of these subassemblies has many unique features that could be discussed in length. However, in keeping with the purpose of this report, only those features that help clarify the system approach, operation and versatility will be discussed here.

The speed of the paper in both the transmitter and receiver can be modified over a wide range by either replacing the drive motors or part of the drive assembly. In many cases, the desired speed can be achieved by simply changing a drive belt and pulley. Thus, the desired paper speed for a particular application can readily be achieved.

For applications that require data rates that are significantly higher than can be transmitted over conditional telephone lines, a broadband single element silicon photodetector is used. The mechanical mounts for both detectors are compatible with the assembly.

The electronics subassembly is functionally designed with unique functions being isolated to individual circuit cards. This permits a specific function to be modified by interchanging the corresponding card. The entire electronics assembly, including the motherboard, can be replaced with another assembly to satisfy those applications with significantly different system specifications.

Although the optics bench was designed so that it can be modified to satisfy a wide range of applications, it possesses a few unique features that enables many requirements to be satisfied with little or no modifications required. The key to the versatility of this facsimile system lies with the optical system design. Therefore, it will be discussed in more detail throughout the remainder of this paper.

Optical System Description

The basic optical system is the same for both the facsimile transmitter and receiver. It is a straightforward coherent optical system design consisting of a 2 mW helium neon laser, an acousto-optical modulator, focusing lenses and a galvanometer. This configuration is shown in the conceptual diagram that is presented in Figure 3 and in the optical system schematic that is presented in Figure 4. A long photodetector is mounted on the outside of the optics bench in the transmitter to detect the light that is scattered off of the document that is being transmitted.

This basic optical system was designed to scan and reproduce photographic documents with a spatial resolution across an 11-inch or less flatbed scan of up to about 300 TV lines per inch within a transmission time of about 1 minute or longer (8.5- by 11-inch document). With 3M Type 7771 dry silver paper as the output medium the system is capable of reproducing up to about 12 shades of gray (IEEE facsimile chart) in the output copy.

As will be discussed later, the spatial resolution can be substantially increased above 300 lines/inch by making relatively inexpensive modifications to the optical system. The transmission time can also be reduced for many applications by utilizing a 6 mW laser that is mechanically interchangeable with the 2 mW laser. More expensive coatings can also be used on the optical elements to further improve the overall efficiency of the optical system. The light intensity at the paper can be increased by using a more efficient AOM than is used in the basic system.

Refer to the optical system schematic shown in Figure 4. The laser beam from the 2 mW laser is projected through the acousto-optical modulator (AOM) and brought to focus on the optical axis by lens L1. This focused spot is reimaged by

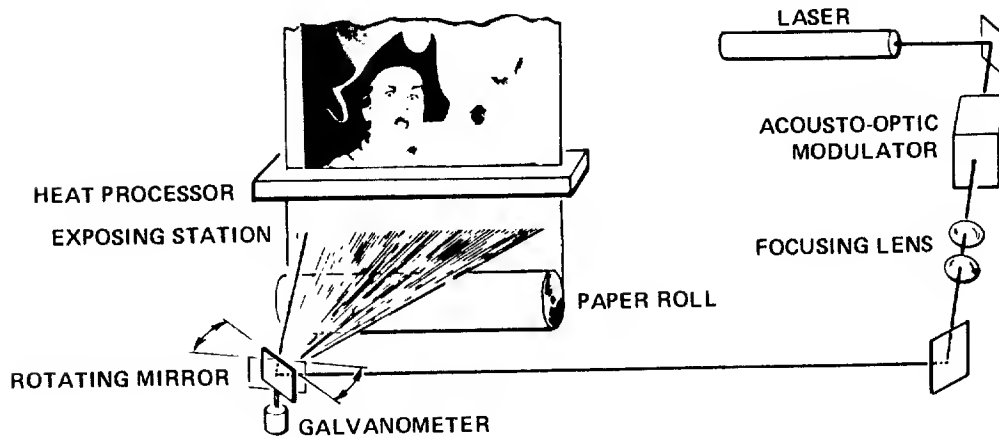


Fig. 3. Laserfax conceptual optical system diagram.

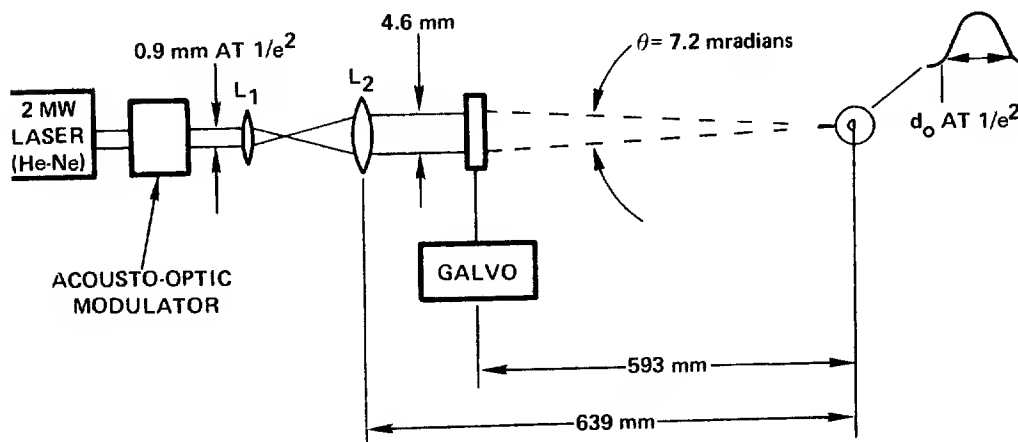


Fig. 4. Laserfax optical system schematic (typical parameters).

lens L2 onto the paper after the beam has been reflected by a mirror that is attached to the shaft of the galvanometer. Stationary mirrors that are used to fold the optics path into a rigid and compact optics bench are not shown in Figure 4. A beam splitter that is located between the AOM and lens L1 is also not shown in this figure. Its function is to reflect part of the laser beam onto a small photodetector to provide a feedback loop to the AOM that is used to normalize the output power of the laser.

In the mechanical package of the optics bench lens L2 is hard-mounted a distance of 639 mm from the paper and the galvanometer is located a distance of 593 mm from the paper. Lens L1 is attached to an axial adjustable mount for two reasons. First, this provides a focus adjustment in the system after the bench has been assembled. The second but most significant reason is that lenses with different focal lengths can be mounted in this location to provide a wide range of spot sizes at the paper. As will be discussed later, this is a significant feature where a high resolution system is to be achieved.

The diameter of the laser beam to the left of the lens L2 is normally about 0.9 mm ($1/e^2$ points). In the system shown in Figure 4 the beam has been expanded to 4.6 mm by the lens combination. The diameter of the focused spot at the paper is therefore

$$d_o = \frac{4\lambda}{\pi\theta} = 0.112 \text{ mm} = 0.0044 \text{ inch.}$$

It is clear that if this spot is used to reproduce photographs of a scan line density of from 100 to 200 lines/inch, a significant roster structure would be present in the output copy. To prevent the roster structure and to provide added versatility to the optical system, the focused laser spot of the paper is deflected perpendicular to the galvanometer scan direction at a very high rate. This high rate vertical scanning is referred to as spot dither and is achieved with the AOM. The drive signal to the AOM is amplitude modulated with the video signal in the receiver but its amplitude is held constant in the transmitter. It is also frequency modulated in both the transmitter and the receiver to achieve spot dither at the paper. Figure 5 shows the effective spot profiles in both the horizontal and vertical dimensions with dither providing spot elongation in the vertical dimension. The parameters presented in the figure are for a typical system with a scan line density of 110 lines/inch.

It is clear that the line height can be decreased by reducing the frequency deviation of the FM signal to the AOM and the line density can be increased by either driving the paper at a slower speed or by increasing the scan rate of the galvanometer. Hence, the resolution of the facsimile system is now electronically controlled. The speed of the galvanometer is directly proportional to the video signal bandwidth. And, the speed of the paper determines the transmission time. These electronically controlled system performance parameters are the trade-offs that provide the versatility that is required to enable a wide range of applications to be readily addressed with this basic facsimile system.

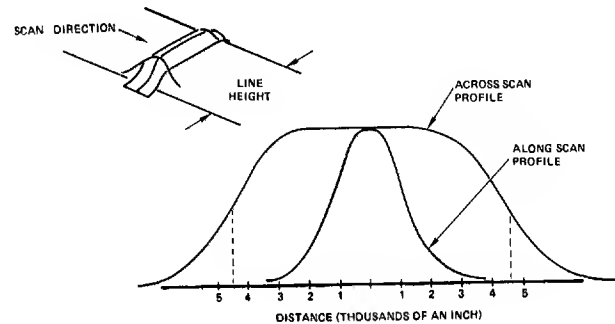


Fig. 5. Typical spot profile for Loserfox system.

AN OPERATIONAL VIDEO DATA COMPRESSION SYSTEM FOR ATS AND ITOS

R. L. Kutz
NASA
Goddard Space Flight Center
Greenbelt, Maryland

L. D. Davisson
University of Southern California
Los Angeles, California

Abstract

An operational data compression system has been developed and implemented for transmission of digitized ATS and ITOS-VHRR satellite video data over the wideband communication link between the Wallops Island, Va. Command and Data Acquisition Station and the National Environmental Satellite Service at Suitland, Md. This system uses minicomputers for the coding and decoding of the data to achieve maximum flexibility together with specially designed interface equipment for greater efficiency. No loss in data quality occurs due to the compression, and, in certain cases, data is transmitted which would be otherwise unavailable due to the limited channel capacity. This paper describes the method of compression, the equipment used, and the compression results attained.

Summary

A real-time programmable data compression system has been developed for operation over a digital wideband communication link between the National Environmental Satellite Service (NESS) Wallops Island, Virginia, command and data acquisition station and the NESS Data Processing and Analysis Division at Suitland, Maryland. The NESS is part of the National Oceanic and Atmospheric Administration (NOAA) of the Department of Commerce. The source data consist of satellite derived images of the earth from which environmental data are extracted by NESS and others. Four spacecraft have been used to provide the source data. The first imagery data used came from the Applications Technology Satellite (ATS) 1 or 3 Spin-Scan Camera and the imagery currently being employed comes from the NOAA 2 or 3 spacecraft Very High Resolution Radiometer (VHRR). The imagery data are received at Wallops Island, Virginia, compressed, transmitted over a 300 km wideband ground link to Suitland, Maryland and reconstructed in real-time. Using general purpose minicomputers together with specially designed input/output channels, a reduction in channel time-bandwidth requirements of two-to-one can be attained with zero loss of data quality with respect to the quantized uncompressed data. As far as the final user is concerned, the system is invisible. Because of the tremendous flexibility of the minicomputers, it is possible to vary the compression algorithm and to perform auxiliary tasks such as data formatting. The compression system has seen continuous use 18 hours a day from December 1972 through July 1974.

Introduction

For many applications, technology has created the ability to generate ever increasing amounts of data. With the generation of this data comes the increased cost of transmission, storage, and processing in proportion to the greater channel capacity required. Data compression is a highly promising technique which can be applied to reduce the increased capacity requirements through more efficient coding of source data. Such techniques have been studied for the last ten years but as yet have had little impact on actual operations due, at least in part, to the increased hardware complexity, lack of appreciation by operational people, and reluctance on the part of users to allow their data to be subjected to any more manipulation than is absolutely necessary. The time has come, however, when economic considerations necessitate the careful review and evaluation of techniques that suggest a more cost effective use of the channel storage and computer capacity, e.g. data compression. This paper describes one important application to an operational situation.

Large quantities of earth-observational data are now generated and it is possible to apply sophisticated processing to that data for many purposes. Principal among these data are meteorological observations for the many aspects of weather prediction with their obvious economic benefits. The use of satellites to acquire images of the earth on an operational basis has dramatically increased the quantity and quality of data involved in weather prediction. The NOAA image data are transmitted from two Command and Data Acquisition (CDA) stations to a central location for computer processing. No processing capability exists at the CDA stations. The particular link used for the data compression system to be described is a microwave link between the CDA station at Wallops Island, Virginia and the Data Processing and Analysis Division of NOAA/NESS at Suitland, Maryland. This link has been used operationally since 1969 for digital ATS spin scan image data without data compression. The recent deployment of the Synchronous Meteorological Satellite (SMS) involves the transmission of higher resolution data than currently installed NOAA/NESS ground transmission equipment can handle. It is apparent that the channel capacity needed to transmit meteorological data will continue to increase as the multiple SMS and TIROS-N spacecraft go into operation.

With the required large channel capacity, data compression provides a method for significant economic savings in transmission costs. Basically, data compression in this application is based on the fact that data from TV pictures of the earth have a significant amount of redundancy from one resolution element to the next. Nevertheless, since it is not known, a priori, where the redundancy will exist within the picture, the data transmission system without data compression is generally designed as if no redundancy exists. In effect the system transmits each resolution element independently. Data compression techniques involve the processing of the picture data to remove redundancy prior to transmission so that a reduced quantity of data is transmitted. Upon reception of the compressed data, processing reinserts the redundancy to reproduce the original image data.

Because of the potentiality of data compression, a feasibility study was initiated by the University of Southern California in participation with and under contract to NASA Goddard Space Flight Center with the cooperation of NESS. Theoretical studies complemented by an empirical analysis of taped digital data verified that data compression could be applied at a significant economic savings to ATS video data. After the NOAA-2 spacecraft launch in October of 1972, the ATS derived data compression algorithm was applied to the NOAA-2 Very High Resolution Radiometer (VHRR) data. A performance comparison verified that the ATS data compression technique could also be used with VHRR data to produce a significant economic savings. A more detailed description of the data, the compression techniques, and the equipment is contained in the succeeding sections.

Data Source

Two types of spacecraft have been used to provide the source data. The first type is the ATS (1 or 3) spacecraft in geostationary orbit — height 35,780 km, period 24 hours — and the second is the NOAA (2 or 3) spacecraft in sun-synchronous polar orbit — height 1460 km, period 115 minutes.

The ATS 1&3 spacecraft are spin stabilized with a spin rate of 100 rpm and with the spacecraft spin axis parallel to the earth's spin axis. The ATS data used is from the ATS spin scan camera which sweeps one image line per spacecraft revolution. The ATS spin scan camera image line period is determined by the spin period of the spacecraft, nominally 600 milliseconds. To permit the full earth disc to be scanned from pole to pole, a worm drive moves the image field of view via the scan optics so that the optical axis stays in the plane containing the spacecraft spin axis. The two mile image field of view is moved southward two miles for each revolution of the ATS spacecraft. A full earth image takes 24 minutes to scan 2400 image lines. Since the ATS 1&3 spacecraft are in synchronous orbit, the earth subtends an angle of 18 degrees or 1/20th of a revolution. The video data are derived from a photomultiplier tube in the spin scan camera which points at the earth for only 30 msec. out of each 600 msec. scan period. The data are transmitted in analog form to the CDA station where it is digitized for transmission to the central data processing location. A digital synchronizing circuit uses a spacecraft derived sun pulse to produce a signal to indicate when the A/D converter should begin to operate for the 30 msec. of useful data out of each 600 msec. spin period.

The sampling rate and digital resolution depends upon the user requirements and the transmission capability. Nominal characteristics are 4096 samples per line each quantized to one of 64 levels and represented by a six bit binary word. The communications link is a 48 khz wideband group with Western Electric 303C modems operated asynchronously without a scrambler.

The NOAA spacecraft are 3-axis stabilized so that the scanning radiometers always point earthward. The spacecraft rotates about its pitch axis once per orbit with the pitch attitude being nominally maintained to within ± 0.5 degrees. The NOAA spacecraft data used with the data compression equipment come from a scanning radiometer called the Very High Resolution Radiometer (VHRR). The two-channel scanning VHRR instrument is sensitive to energy in the visible spectrum (0.6 to 0.7 microns) and infrared window (10.5 to 12.5 microns). Within the VHRR instrument, energy is gathered by a 12.7 cm elliptical scan mirror, set at an angle of 45 degrees to the scan axis, and a telescope. The scan mirror rotates at 400 rpm which causes the half nautical mile image field of view to sweep the earth in a direction perpendicular to the NOAA spacecraft ground track once every 150 msec. One image scan line is generated per revolution of the VHRR mirror. The consecutive scan lines of the VHRR data can be used to produce an image due to the spacecraft motion along the orbital track which causes the subsatellite point to move one half nautical mile per 150 msec. — one scan line period. The VHRR instrument produces a continuous stream of scan line image data without frame segmentation. Since the NOAA spacecraft are in a low altitude orbit compared to ATS 1&3 the earth is viewed for one third of each VHRR scan or about 50 msec. The VHRR data are handled in two ways. First, the VHRR data are continuously transmitted throughout every orbit. This mode of operation is known as the High Resolution Picture Transmission (HRPT) and at the ground station the data are designated HRPT data. Second, in addition to the HRPT mode of operation up to 8.5 minutes of data may be recorded onboard the spacecraft for later playback when commanded during a CDA contact. The spacecraft recorded VHRR instrument data are designated VREC data. The VREC data are played back from the reel-to-reel spacecraft recorder by driving the tape in the reverse direction and at the same speed as that used for recording. At the CDA station the VREC data differ from the HRPT data in that the time axis is reversed, i.e. the VREC data are backwards with respect to the HRPT data. Tape speed variations affect both timing and amplitude of the playback VREC data (FM recording is used), however the timing error is much more noticeable in the displayed image without compensation. Image compensation is provided for by recording a reference tone along with the VREC instrument data on the spacecraft recorder. During playback the reference tone is transmitted on a separate subcarrier to permit its use on the ground in compensating for tape recorder flutter and wow (F&W). The space-to-ground NOAA-2&3 communications link uses FM-FM modulation. At the CDA station the NOAA spacecraft data are demodulated to produce FM modulated HRPT data, VREC data, and the VREC F&W reference signal. These three signals plus a CDA crystal controlled reference tone are recorded on an instrumentation ground recorder in the 'direct mode' at 60 inches

per second (ips). All of the NOAA-2&3 data processed by the data compression system is obtained from the CDA instrumentation recorder operating at one fourth the record speed or 15 ips. The VREC F&W signal passes through two record/playback steps and the HRPT F&W signal through one step since it is generated locally at the CDA station. No attempt is made to correct the amplitude variations introduced in the FM modulated HRPT & VREC data signals due to the F&W of the analog tape recorders. However the F&W signals are used to control the sampling rate of the analog to digital (A/D) converter and this corrects for timing variations. The A/D converter is part of the data compression equipment and is preceded by an FM discriminator, figure 1.

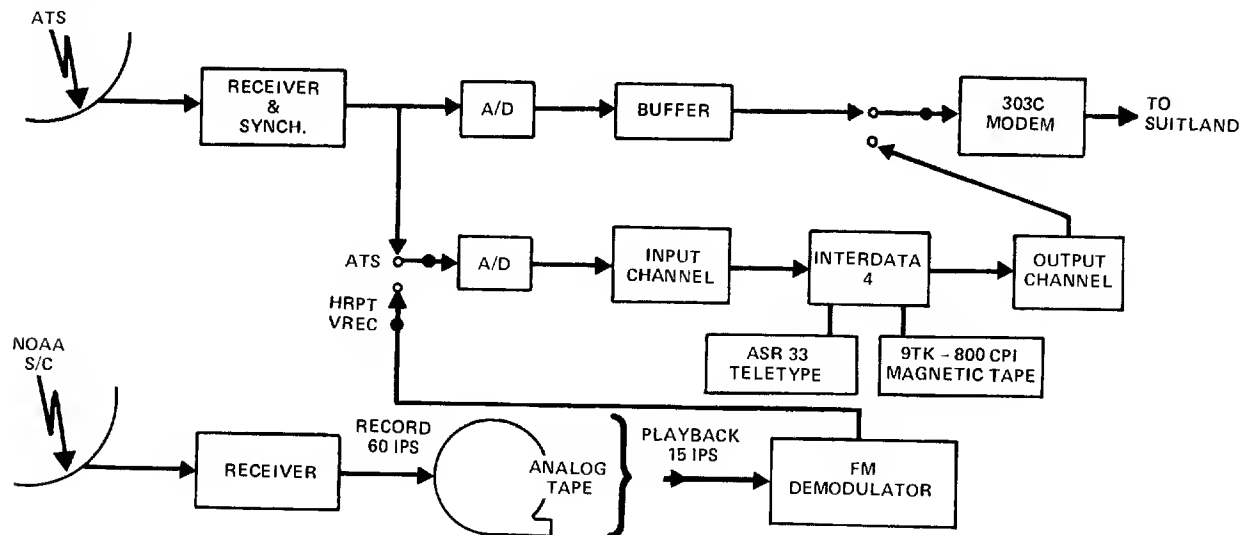


Figure 1. Wallops Island Data Compression System

Implementation Requirements and Constraints

In the case of the ATS data compression experiment an operational digital transmission system had been installed in 1969 and was in daily use between the Wallops CDA station and the Suitland NESS central data handling facility when the data compression experiment was conceived. At Suitland the ATS data is used to produce hundreds of wind vectors over the oceans on a 400 km grid to be used as one type of input to the primary NOAA numerical weather model. The ATS data is also converted to an analog facsimile format which is transmitted over 4 khz phone lines with C5 conditioning to remote weather analysis centers e.g. the hurricane watch center in Miami, Florida and the tornado watch center in Kansas City, Kansas. In an on-line weather data transmission system, particularly where the data is being used to monitor severe storms, reliability is very important. For this reason, it was necessary to test the data compression system in an off-line mode before it could be used to transmit real-time data in an operational mode. Since the non-data compression operational ATS Data handling system between Wallops and Suitland handles digital data quantized to six bits per sample in a gray code, it is necessary to deliver reconstructed data from a data compression system in the same format. NESS also requested that the data compression system not be allowed to introduce errors in the samples due to the compression algorithm. The data compression system is required to be transparent to the receiving terminal at Suitland and to the data user. To provide for ease of testing and rapid recovery in the event of a malfunction, the data compression system is designed to be inserted or removed from the transmission system by the operation of a single switch at each end of the wideband communications link. The operation of the data compression system is designed to be simple to insure that operator errors are minimized. Several versions of the software interface between the operator and the data compression hardware were required to achieve a satisfactory level of operation.

The NOAA spacecraft data can be effectively handled by preprocessing to provide an improved line synchronization, calibration, and gray scale correction. The image display device used with the NOAA spacecraft data is the same as that used for the ATS data and all users on the ATS facsimile circuit can receive NOAA 2&3 VHRR images. Since it is desired to produce a 1:1 aspect ratio in the center of the VHRR images and the line to line spacing is fixed by the worm drive in the photofax display, it is necessary to select a sampling rate which allows 3/4 of the earth view to be displayed. This is a satisfactory compromise since the geometric distortion of the image is significant beyond the area displayed, due to the low 1460 km orbit of the NOAA spacecraft. The photofax display available for the VHRR data is the same one used to display ATS image data and operates at 100 rpm. At the CDA station the VHRR data are received at a 400 line per minute rate and recorded on a Mincom instrumentation recorder at 60 ips. It would have been preferable to digitize the VHRR data at the CDA station as it was received from the spacecraft, however the ATS data compression hardware cannot handle the high data rates necessary. A high speed large volume digital data buffer would be required and funds were not available for its procurement. The analog Mincom recorder is used to replace the digital buffer and to slow down the VHRR line rate from 400 to 100 lines per minute by playing back the tape at 15 ips. Two types of operations are required for the VHRR data handling; the first type transforms the VHRR data into

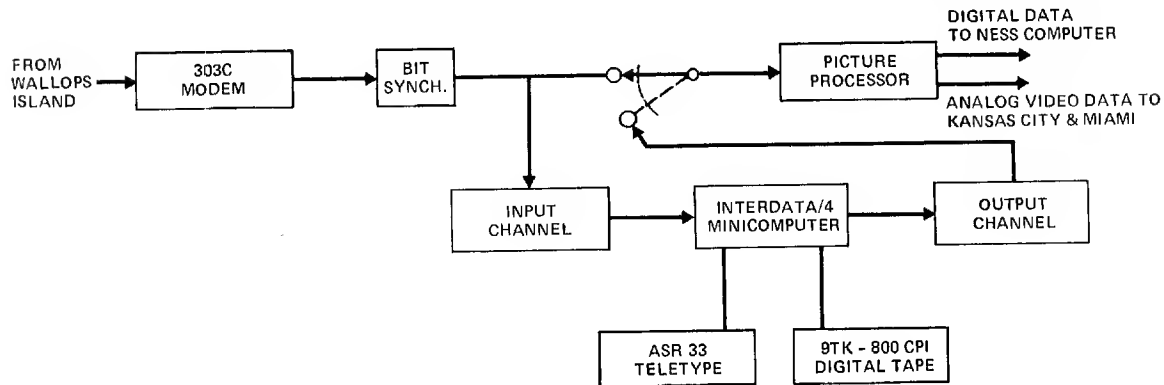


Figure 2. Suitland Data Reconstruction System

a format compatible with the ATS digital display and data handling equipment. This has the advantage of requiring only half of the processing equipment — the Wallops end of the compression system. The second type of operation required is that digital VHRR data be quantized to 8 bits and transmitted to Suitland. An additional requirement of the second type of operation is that 100% of the VHRR scan be reconstructed at Suitland while only 50% was needed in the first type of operation. Since the use of data compression could accomplish the data volume objectives of the second type operation, it was further required that no error be introduced in the reconstructed data due to the compression algorithm in the absence of channel errors.

Data Compression Techniques

The goal set for the data compression system is to attain a reduction in transmission rate of two to one with six bit ATS data or eight bit VHRR data and not introduce error due to the compression algorithm. The algorithm used has been chosen to meet the real-time requirements for error free compression within the limitations of the Interdata 4 minicomputer used at each end of the compression system. The data compression algorithm can be varied within wide limits because of the use of general purpose minicomputers to do the encoding and decoding. Presently the algorithm forms successive sample-to-sample differences and encodes the differences of up to a maximum magnitude of 9 for six bit data or 14 for eight bit data using a variable length code. When the difference exceeds the maximum magnitude, the sample value itself is sent with an appropriate prefix to distinguish this from a difference code word. The code words are chosen using the Shannon-Fano encoding procedure with empirical frequencies for the relative probabilities. In particular, a difference greater than 14 is encoded by sending a prefix of four zeros followed by the sample from the data.

The above procedure was found to result in a code whose efficiency is greater than 90%. The code words are characterized by a prefix whose leading zeros identify each code word's length and decoding table.

When redundancy is removed, the data quality becomes more sensitive to channel error. To combat this effect, controlled redundancy is added in the form of synchronization words. In addition to the usual line synch., data synch. words are periodically inserted in the encoded data after each fixed number of encoded samples. Following each synch. word the difference from zero is sent rather than the sample difference to eliminate any error accumulated at the receiving terminal. Each scan line starts with a 32-bit line synchronization word, which establishes both line and word synchronization. The 4096 data samples in each scan line are next encoded into 16 blocks with each block representing 256 data samples. At the end of each block is a 16-bit data synchronization word which ensures that word synchronization is maintained and that word synch. can be reacquired within 256 data samples in the event of a synch. loss. Due to the variable length coding, the number of bits between two successive synchronization words varies. Although the data compression encoding procedure selected leads to a data format where word synchronization depends on a low probability of channel error, experience has shown that the data compression system has a more robust synchronization characteristic than an image transmission system in which there is only one line synchronization word on a scan line and in which word synchronization is obtained from a fixed word length data format. The variable block length and hence variable line length data format produces a variable number of bits per image scan line; the number of bits depends on the redundancy in the scene at the scan line location.

In figure 3, the image scan line number runs from 0 to 2000 on the abscissa, and the ordinate indicates the percentage of non-redundant data from the original uncompressed line remaining to be transmitted after data compression. The image data to which figure 3 corresponds is an ATS spin-scan cloud cover image of the full earth disc enclosed by a black space-view surround. Figure 3 shows that more data must be transmitted in the middle of the picture than at either end, since the intersection of the scan line with the earth disc has its maximum chord length near line number 1000. However, not all of the data compression in each line is due to the constant amplitude space-view, since space-view accounts for only 10 percent of the uncompressed data near line 1000 and yet the compression ratio exceeds two-to-one in this area of figure 3. The area above the curve of figure 3 represents the increased capacity of the data compression communication system as a function of the ATS image scan line number.

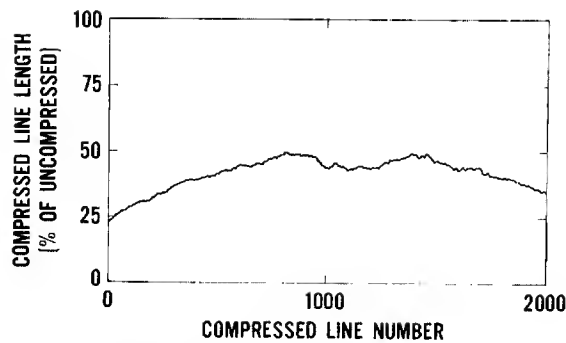


Figure 3. % Data Remaining After Data Compression on a Line-by-Line Basis

ENTER MODE (0 - 20):XY				
X		Y	TAPE	OUTPUT
0	ATS	0	OFF	COMPRESSED
		1	OFF	GRAY
1	HRPT	2	RECORD COMP.	OFF
		3	RECORD GRAY	OFF
		4	RECORD 8 BIT	OFF
2	VREC	5	RECORD 8 BIT	GRAY
		6	RECORD 8 BIT	COMPRESSED
		7	READ 8 BIT	GRAY
		8	READ 8 BIT	COMPRESSED
		9	READ { GRAY OR COMP.	GRAY
		A	RECORD COMP.	COMPRESSED
		B	RECORD GRAY	COMPRESSED
		C	RECORD QUANTITATIVE	GRAY
				OFF

Figure 4. Wallops Commands

Since data compression can produce short term increases in data quantity, the shorter of the compressed image line and the original image line without compression can be transmitted with a special code used to indicate which. In spite of the obvious varying data activity from one pole of the earth to the other in figure 3, the compressed line length remains relatively constant suggesting that the system can be quite effective with only a one line buffer. This is in fact true except for noisy scan lines which are usually of little interest to the user.

Compression System Hardware

The Wallops Island data compression equipment configuration is shown in block diagram form in figure 1. The front end of the system accepts analog data from one of three sources — ATS, HRPT, and VREC, of which the latter two are derived from the NOAA-2&3 spacecraft VHRR instrument. The analog data are sampled, quantized to eight bits and stored in the Interdata-4 (I4) core memory through a direct memory access channel. The A/D sampling rate is controlled from the I4 by loading a register with the desired sampling rate control word. Between the A/D converter and the I4 memory is the A/D and direct memory access control logic. The sampling rate of the A/D converter may be phase locked to the F&W signal from the CDA ground recorder or a crystal controlled oscillator. The input direct-memory-access (DMA) logic may be triggered to begin storing A/D samples in memory. An externally derived line synchronization pulse is used for this purpose.

There are four DMA channels in the data compression system, two at Wallops Island and two at Suitland. At Wallops and Suitland one DMA channel is used for input and one is used for output. A DMA channel has a 16 bit register which may be loaded by an instruction from the I4. This register is called the Pointer Register (PR). The PR must be loaded with the I4 memory address of the DMA channel instructions. The DMA channel instruction list contains three types of information, the start and end of an I4 memory data I/O buffer, the address of the next instruction list, and DMA channel control bits to indicate the action of the channel when an I/O buffer data transfer is complete. The starting and final address registers of the DMA channel are loaded by the DMA control logic from the I4 core memory instruction list once the 'DMA START' instruction is executed by the I4. Once started the DMA channel requires no further action by the I4 CPU and thus, leaves the CPU free to compress data, etc. The DMA channel can generate interrupts when each I/O buffer is complete. This permits the I4 computer program to maintain synchronization with the I/O to be processed. It is a simple matter with this type of DMA channel to generate interrupts at the end of several regions of an input scan line and initiate different types of processing algorithms for each region. This procedure is used to control the timing of the data compression/reconstruction software.

There are three other elements of the data compression hardware which are used in both the Wallops Island redundancy removal equipment, figure 1 and the Suitland data reconstruction equipment, figure 2: an ASR 33 teletype, a 9 track 800 CPI digital magnetic tape drive, and the 16K byte (eight bit) core memory of the I4 minicomputer — 1K = 1024. The teletype is used to transmit operator commands to the compression system software and to display the compression system status both for the operator and as a historical record for future reference. Programs are entered into the I4 memory from the digital 9 track tape. Simple revisions to the software are made while the program is core resident by modifying, adding, deleting the necessary I4 instructions/constants, and finally writing the modified core image out to the digital tape for later re-entry. The 800 CPI 9 track tape runs at a speed of 25 inches per second which produces a data transfer rate of 20,000 bytes per second. The transfer of data between tape and the I4 memory is handled by a standard Interdata Selector Channel which operates as a DMA interface between the tape controller and the core memory. The Interdata Selector Channel must be set up under program control of the I4 and can transfer only a single sequential data area before it halts and must be set up again. Major changes to the compression system software are accomplished at GSFC or USC, stored on digital tape, and sent through the mail to be used at Wallops Island or Suitland. However, the main use of the digital tape is for the temporary storage of processed data. At Wallops the digital tape is used for recording when the Wallops-Suitland communications link is unavailable. The advantage of having the data on digital tape is that for the VHRR data the image data stream can be easily broken into 2400 line frames with a 100 line overlap. The overlap consistency is difficult to maintain when working from analog tape.

UNDERLINE Indicates Operator Input

<pre> B.V. 17.4 3/28/74 ENTER MODE(0-2C):<u>10</u> MODE 10 VHRR TAPE OFF D* <u>040974</u> T* <u>1900Z</u> R/O* <u>1901</u> VIS OR IR* <u>VIS</u> 040974 1900Z VHRR 1901 VIS TELE ON SEND COMP * <u>GO</u> GO * <u>L=100</u> L=100 * <u>L=1000</u> L=1000 * COMP. RAT.=2.92 END 1000 LINES </pre>	<pre> * <u>ST</u> ST B.V. 17.4 3/28/74 ENTER MODE(0-2C):<u>10</u> MODE 10 VHRR TAPE OFF D* <u>040974</u> T* <u>1935Z</u> R/O* <u>10-908</u> VIS OR IR* <u>IR</u> 040974 1935Z VHRR 1908 IR TELE ON SEND COMP * <u>GO</u> GO * <u>L=100</u> L=100 * COMP. RAT.=2.15 END 0100 LINES </pre>
--	--

Figure 5. Example of System Control Teletype Input-Output



Figure 6. ATS-3 Spin Scan IMAGE 17, Dec. 21, 1972

The Wallops CDA compression hardware contains an output parallel-to-serial converter to interface the I4 memory to the 303C modem. The output bit rate is selected from the I4 software by loading a register in the output channel control logic. Between the I4 core memory and the output controller logic is a DMA channel described above. Note that none of the compression operations are done with special purpose hardware.

The NESS data reconstruction equipment at Suitland is similar to the Wallops Island system. However the input and output device controllers are different. The input device controller receives a clock and serial data from a bit synchronizer which is in turn driven from the receiving Western Electric 303C modem at Suitland. The input channel logic can be set up to find both the 32 bit line synch. and the 16 bit data synch. using a Hamming metric with the allowed distance usually set to zero. The input channel is essentially a serial-to-parallel converter with two selectable modes of operation provided for handling the serial input data. The first mode assembles sequential 16 bit words of data from the incoming data stream and sends the parallel words to the DMA channel to be stored in the I4 core memory. The second mode uses a hardware decoder to help reconstruct compressed data. The code class which can be decoded by the hardware must have a prefix of N leading zeros terminated by a one, N is from 0 to 15. The hardware compression decoder has been used to reconstruct six bit source data. Implementation of the hardware compression decoder was undertaken to speed up the data reconstruction process. It has since been found that the pure software reconstruction algorithm for eight bit source data could be written to operate nearly as fast as the six bit data hardware/software decoder combination. The hardware decoder, as implemented, has a disadvantage in that it stores 16 bits in the I4 memory for each sample decoded, producing a temporary data expansion, and does not allow compressed data to be buffered since it is not available in the core memory.

At Suitland the output device controller is the final element of the data compression communication system. The output device controller is basically a parallel-to-serial converter which can produce a data stream in the same format as the Suitland bit synchronizer. The Suitland output device drives the NESS digital picture terminal which is the distribution point at Suitland for analog facsimile data to be used by remote display devices as shown in figure 2. The NESS digital picture terminal also generates images for use at Suitland and sends digital data to the NOAA weather computer, an IBM 360-195. The output channel requests data from a DMA channel connected to the I4 memory. There are two selectable modes of operation for the output channel. First, each eight bit byte can be converted to the six bit gray code format required by the digital picture terminal. Second, consecutive 16 bit halfwords from the I4 memory are shifted out of the output channel device in one long serial stream. The second mode is used to test the Suitland compression equipment input channel. The input device and output device are designed to be driven from independent data clock rates, although only a common clock has been used to date.

Data Compression System Operation

To operate the data compression system the operator initiates execution of the compression system executive software and receives a response defining the program revision number, the date revised, and a request that the operating mode number be entered as shown in figure 4. The operator enters a two character operating mode on the teletype console. The first digit indicates the data source, 0 = ATS, 1 = HRPT, and 2 = VREC. The second digit selects the I/O devices and the type of processing to be used. The operating system responds by requesting that documentation information be entered by the operator. For example, D*, T*, R/O*, VIS OR IR* indicate the date, time, spacecraft orbit, and whether the data is visible or infrared — see figure 5. The VIS OR IR input is used both for documentation and to select the appropriate display gray scale lookup table. Messages are displayed to indicate the status of the input/output devices, e.g., TAPE RECORD 8 BIT, TELE ON, and SEND COMP. When the operator

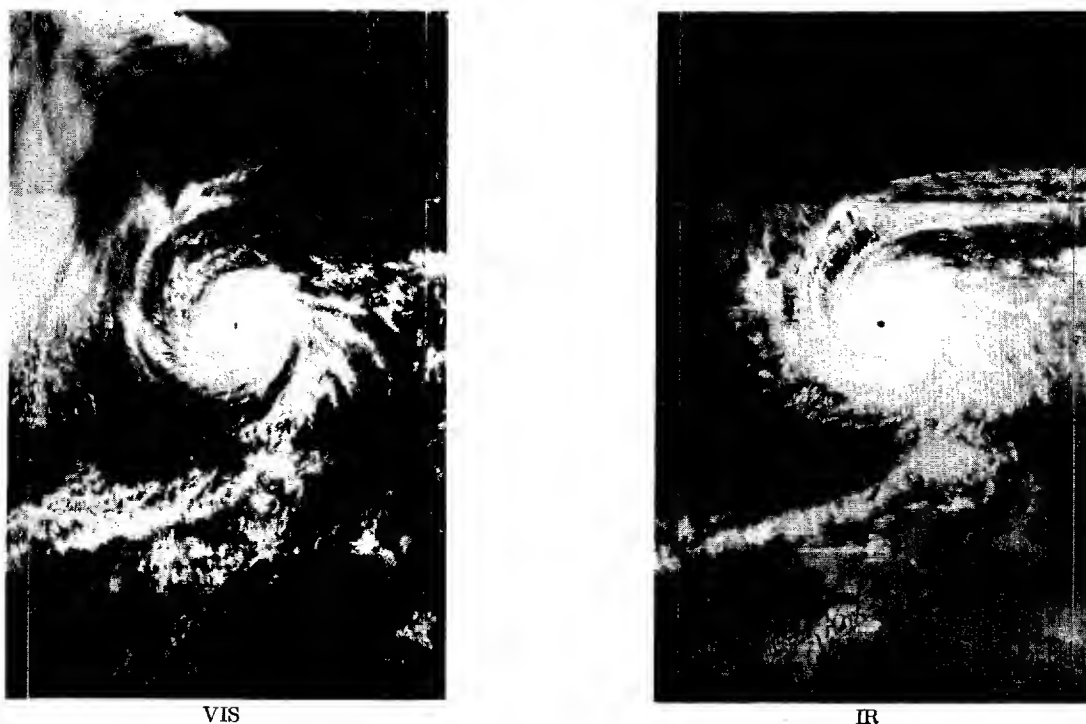


Figure 7. NOAA — 2 VHRR Images of Hurricane "AVA" Rev 2942, June 7, 1973

is ready to begin processing data he can type GO to start or SE to enable the processing of data which will start upon the receipt of a frame synch. signal. At Wallops the termination of the processing of image data by the data compression hardware can occur in two ways — either the line count reaches a preset limit or the operator types FI to finish processing. The line count limit is automatically set to 2400 by the compression system software before the processing of each image but can be changed by the operator command L=XXXX, e.g. L=1000. On the front panel of the Interdata-4 computer are two rows of display lights. Each row of display lights contains four hexadecimal light emitting diode display characters which can be loaded under program control. The upper and lower rows of the I4 display are loaded with the decimal line count limit and the current line count respectively. At Suitland the end-of-image is found by reaching the preset line count limit or by receiving a frame-complete synch. word from Wallops.

At Wallops two types of operating procedures are employed. When the ground link to Suitland is available data are sent in the desired format. When the ground link is busy with other traffic, analog data is input to the compression system and stored on digital tape for later playback to Suitland. These digital tapes are selectively sent to GSFC or USC for evaluation. At Suitland the data compression system is used in two ways. First, to receive, process, and record on digital tape data to be used in the IBM 360-195 operational weather data processing system. Second the data reconstruction system can drive the digital picture terminal and its associated facsimile remote displays.

Results

The Wallops Island, VA to Suitland, MD data compression system has been operating for the past two and one half years, has demonstrated a high degree of reliability, and has helped to achieve a high degree of user confidence in the value of compressed data. The compression ratio varies depending on the image scene content but a typical value is 2:1. The quality of the received and reconstructed data have been monitored in two ways. First, based upon a visual review, the images produced have been found to be of equal or superior quality to those produced without data compression — see example in figures 6 & 7. In the rare cases where a channel error affect is discernable in a reconstructed data compression image, the effect shows up as a streak of less than 1/16 of the image scan line length. Second, an error detection code shows less than one percent of the reconstructed image lines are in error. These image line errors are caused by channel errors and not by the compression system hardware or software.

The role of data compression is not yet clear in the operational NOAA/NESS spacecraft program. The Alaska CDA to Suitland, MD communication system will very likely be reconfigured to take advantage of the economies inherent with communication satellites for long ground distances. Satellite-to-satellite communication systems are being planned which will reduce the necessity for on-board storage and allow continuous real time transmission of spacecraft sensor data to the central data processing facility. Perhaps the ground storage and processing areas will be the first to benefit from the practical application of data compression techniques.

Acknowledgement

The authors are pleased to acknowledge the encouragement, operational support, and helpful comments of R. Koffler, J. Leese, C. Bristor, E. Connelly, and H. Van Dyke of the NOAA/NESS.

A CAQ BANDWIDTH REDUCTION SYSTEM
FOR RPV VIDEO TRANSMISSION

James J. Pearson
Lockheed Palo Alto Research Laboratory
Palo Alto, California

Abstract

A bandwidth compression system for the transmission of video images from remotely piloted vehicles has been built and demonstrated. Novel features of this system are the use of the Constant Area Quantization (CAQ) technique to obtain spatial bit rate reduction of 6:1 and a rugged and compact scan convertor, based on a core memory, to accommodate temporal frame rate reduction. Based on the ability of the human eye to perceive more detail in high contrast regions than in low, the CAQ method transmits higher resolution in the former areas. The original six-bit digitized video is converted to a three level signal by the quantizing circuit and then Huffman - encoded to exploit its statistical properties and reduce it further to one-bit per pixel. These circuits operate on one line of the picture at a time, and can handle information at full video (10 MHz) rate. The compressed information when received on the ground is stored in coded form in a two-frame (500,000 bit) digital core memory. One frame of the memory is filled while the other is being displayed and then the two are interchanged. Decoding and reconstruction of the video are performed between the memory and the display.

Introduction

The use of remotely piloted vehicles (RPV's) to perform tactical reconnaissance and strike missions under conditions too dangerous to permit the risk of sending manned aircraft is being given increasingly serious consideration by the armed services. The eyes of the absent pilot are replaced by transmission of video imagery from an onboard camera, and the success of the mission depends crucially on the integrity of the video link. Video signals are inherently broad band, and the necessity of protecting them from jamming by some form of spectrum spreading technique makes even greater demands on the limited available spectrum. When transmissions from a number of vehicles are occurring simultaneously, as envisioned in most RPV battle scenarios, the spectrum required is simply unavailable. Therefore, some means must be employed to reduce the bandwidth of the video signal prior to transmission.

Fortunately, a sequence of meaningful video images contains enough redundancies to permit its compression by a sizeable factor without the loss of significant information. These redundancies are of two basic types--temporal and spatial. The temporal, or frame-to-frame, redundancies are most practically exploited by reducing the frame rate from the standard thirty frames per second to some lesser number, dictated by the particular mission requirements, through the use of some form of slow-scan camera. The spatial, or single frame, redundancies arise from the existence of finite range correlations within the picture. It is in the choice of a method to exploit these spatial redundancies that the constraints of the RPV application are most strongly felt. Since the vehicles are small and light, often comparable to large model airplanes, the equipment must be compact and lightweight and require little power. In addition, since the RPV's are, to some degree at least, expendable, the equipment must have a low cost.

With this application and its attendant constraints in mind, the Lockheed Palo Alto Research Laboratories have designed, built and demonstrated in the laboratory a bandwidth compression system based on the Constant Area Quantization (CAQ) technique. The Lockheed system performs spatial compression to reduce a six-bit per pixel digitized input signal to one-bit per pixel. It is designed to operate either at full video rate or at any reduced frame rate for which the jerkiness and delay are tolerable in the particular application. In the reduced frame rate case, scan conversion is performed on the ground in a rugged and compact digital memory, and an analog signal is delivered to a standard monitor at thirty frames per second.

Principle of Operation

Constant Area Quantization is a one-dimensional spatial compression technique. It is based on the property of human vision that the eye sees more detail in high contrast regions than in low contrast ones. This property is illustrated in Fig. 1, where the circles decrease in size from left to right and in contrast from top to bottom.

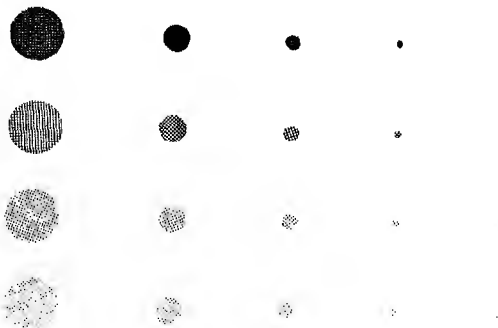


Fig. 1 Test pattern demonstrating the principle of Constant Area Quantization.

If the figure is viewed under conditions of distance and illumination such that some but not all of the circles are visible, it will be seen that the cutoff is not along a verticle or horizontal line but along a diagonal, indicating a combination of size and contrast. This fact is exploited by transmitting higher resolution in the high contrast areas where the eye will detect it, and lower resolution elsewhere.

In practice, this is accomplished in the simple way shown in Fig. 2. The bold line in 2 (a) represents the analogue video signal which is sampled at a rate indicated by the tick marks. The smooth curve is replaced by line segments, beginning and ending at sample points, in such a way that the right triangles with these segments as hypotenuse and sides parallel to the coordinate axes all have the same area, $A/2$. The value of the area threshold A , is an arbitrary parameter chosen at the outset. It has the somewhat peculiar units of brightness levels times pixels. At each sample point the output signal in 2 (b) is generated. This signal is zero except at the end of a line segment, where it is either +1 or -1 (P or N) depending on whether the signal increased or decreased during that interval. In this way, a six-bit (64 level) input signal is reduced to a three level one. In binary terms three levels corresponds to $1.58 (= \log 3 / \log 2)$ bits, and this value can be approached arbitrarily closely by coding a number of picture elements together. (For example, the 243 possible values for five adjacent pixels can be represented by one eight bit number, giving an average of 1.6 bits per pixel.) The original picture can be reconstructed from the three level signal in two different ways illustrated in 2 (c) and 2 (d). In the first or "slope" method, the line segments in 2 (a) are reconstructed from a knowledge of the base and area of the appropriate right triangle. In the second, or "step" method only the endpoints of the segments are computed and the intervening pixels are filled in with constant values. The relative performance of the two reconstruction techniques is discussed later, but the step reconstruction is used in the actual system.

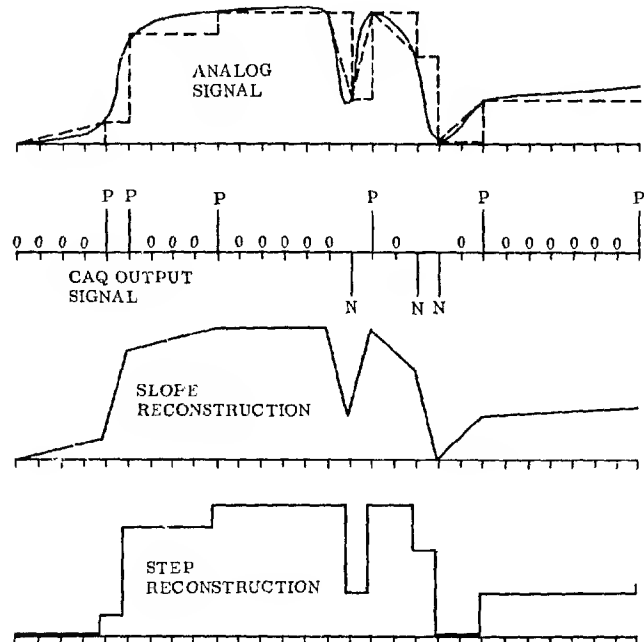


Fig. 2. Constant Area Quantization applied to a simulated video signal.

The way in which the three level signal is generated is shown schematically in Fig. 3. The video signal is sampled and digitized to form the signal V_1 for pixel i . V_1 is subtracted from the reference level R corresponding to the beginning of a line segment in Fig. 2 (a). The magnitude of the resulting difference is then compared to the value A/Δ_1 , where Δ_1 is the time difference between pixel i and the pixel at the beginning of the segment. If $|V_1 - R| \geq A/\Delta_1$ this indicates that the area, $1/2 |V_1 - R| \Delta_1$, of the

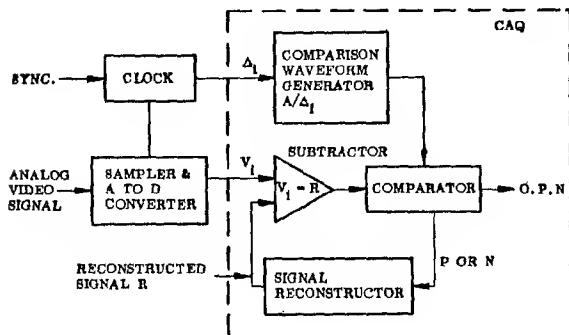


Fig. 3. Constant Area Quantizer block diagram.

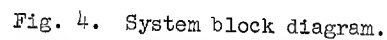
previously mentioned right triangle exceeds the value $A/2$. When this occurs, a P or an N is produced depending on the sign of $V_1 - R$; otherwise a zero is output. Whenever a P or an N is generated, a new reference level, R , is formed by the reconstructor, and a new line segment is thus initiated. The value R , which is identical to the picture element which will be reconstructed on the ground, differs from the actual video value, V , because the threshold is in general not reached exactly at a sampling point. R is used instead of V to prevent the accumulation of errors.

Experience with a number of pictures indicates that the three output signals of the CAQ are not equally probable. In fact, zeroes predominate to such an extent that a considerable additional bandwidth compression can be achieved by using some form of statistical coding. The code actually chosen is a Huffman code, applied to the nine possible values of a pair of adjacent pixels. The code itself is given in Table I. It is a variable length comma-free code, which assigns the shortest code (0) to the most probable event (two adjacent zeroes) and the longest codes

System Description

The bandwidth compression system, as it would be configured for flight is outlined in Fig. 4. The analog video output from either a standard or slow scan video camera is sampled and digitized to six-bits (the rest of the system can handle eight-bits if sufficient camera quality and A/D convertor complexity are justified).

The digital signal is fed to the Constant Area Quantizer, which outputs one of three values (0, P, M) at the pixel rate. This three level signal goes to the Huffman encoder whose output comes in bursts of from one to six-bits at the end of each two-pixel interval. This coded signal must be buffered for uniform transmission. The signal at the output of the buffer has a bandwidth corresponding to one-bit per pixel and can be transmitted over any desired form of digital anti-jam data link.



The received and demodulated signal is stored in coded form in one-half of a two frame memory. While one-half of the memory is being filled, the other half is being used to refresh the display at a full thirty frame per second TV rate. Then the roles of the two halves of the memory are interchanged. If a standard rate video camera is used in the vehicle, this scan conversion step is by-passed. One line at a time is removed from the memory at video pixel rate into a buffer capable of delivering it to the decoder in high speed bursts. The decoder removes from the buffer as many bits as it finds necessary to generate two-pixels in CAQ form. The three level output of the decoder, which is uniform in time, is used by the re-constructor to generate a digital video signal which passes through a D/A convertor to the monitor.

It is not the intention of this paper to describe the circuitry in detail. A few comments about some of the individual blocks in Fig. 4 may, however, prove helpful. All of the circuitry is designed to operate at a 10 MHz pixel rate, to enable it to handle standard video frame rates easily. This speed requirement can be relaxed for the airborne portions of the system if a slow scan camera is to be used, and a simplification in the circuitry results. The ground-based decoder and reconstructor must operate at the high speed, however, since they operate between the scan converter and the display in order to permit storage of the images in compact coded form. Since several operations must, in general, occur within one pixel time, a 40 MHz clock is used throughout. All of the components are Schottky series TTL except in the Huffman encoder, where some 10K series ECL chips are used. The quantizer operates as described in Fig. 3. The quantity $1/4_i$ is generated by a hyperbola generator consisting of a counter and a decoder network. Thresholds are restricted to powers of two so that multiplication by A implies simply a shift in the bit connections. The encoder straightforwardly generates the bits specified by Table I for each pair of pixels, using a series of

NOR gates, and shifts out the appropriate number of them into the buffer. Figures 5 and 6 are photographs of the CAQ and Huffman encoder circuit boards. The decoder is a sequential logic network which takes bits from the buffer until it recognizes the end of a code and then generates the appropriate pair of pixel values. The scan convertor consists of two 32,768 word by nine bit LEC memories on a single chassis. The memories can deliver nine bits to the line buffer in 800 nanoseconds, which is sufficiently fast to assure that 512 bits can be removed in each video line time. This form of scan convertor has the advantage of being compact, rugged, and relatively inexpensive, without adding additional degradation to the imagery.

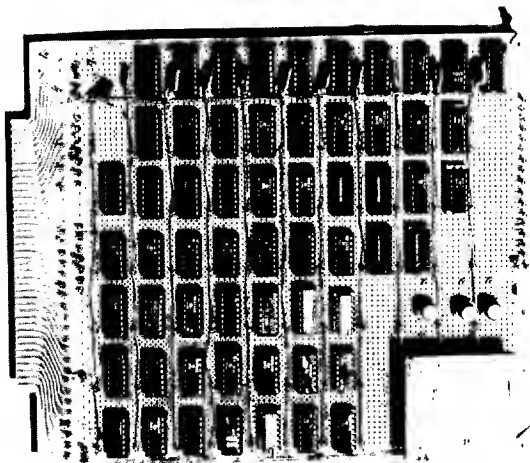


Fig. 5. Constant Area Quantizer Circuit Board.

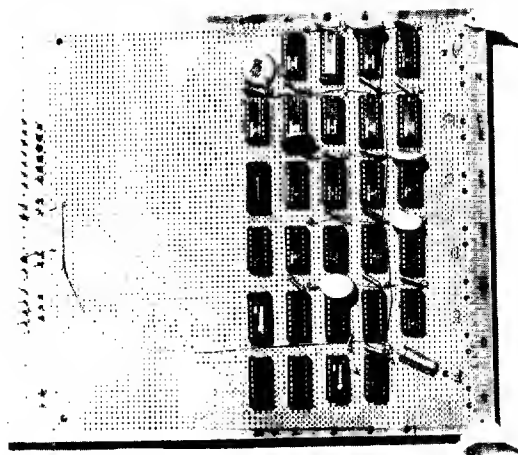


Fig. 6. Huffman Encoder Circuit Board.

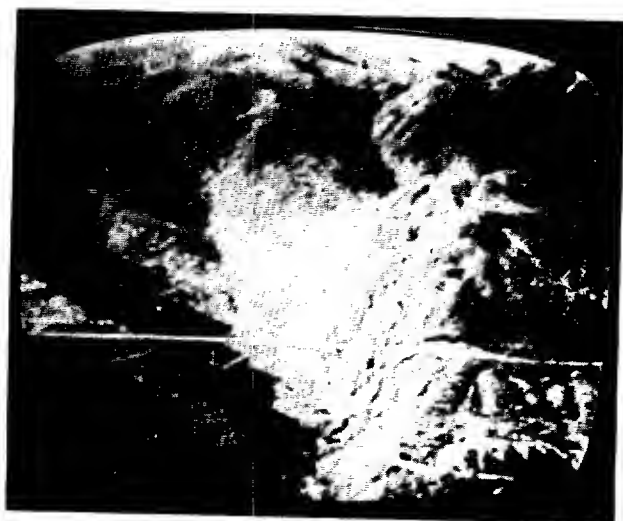
Performance

Since this system is designed to transfer certain types of information from moving video imagery to human observers, the only really satisfactory measure of its performance is its effectiveness in doing that job. Within the confines of a paper and in the absence of any definitive subject tests, more indirect criteria must be used. Most of the pictures in this section, illustrating as they do a wider variety of algorithms and parameter values than are wired into the actual system, are computer simulations. They are all of the same subject to facilitate comparison. The original is shown in Fig. 7. The tank farm subject, having a great deal of detail, is in some sense a "worst case". The simulations reproduced on a microfilm plotter have 256x256 pixels and thus represent only a portion of the actual frame, which in the system contains 480x480 pixels. Figures 8 (a) and 8 (b) are original and compressed versions respectively of video taped imagery run through the actual system and photographed with a Polaroid camera directly off the monitor. The principle parameter which can be varied in the CAQ quantizer is the area threshold, A . Varying A does not alter the amount of bandwidth reduction directly (the output of the quantizer is always 1.6 bits per pixel, independent of both A and the input picture) but it does greatly affect the image quality. A small value of A causes P's and N's to be produced more often, thus increasing the low contrast detail, but reduces the speed with which high contrast changes can be followed, resulting in edge blurring. A large A improves the edge response, but washes out the low contrast detail. The visible result of this is a streaked appearance, as large regions are replaced by constant values which vary from line to line. Fig. 9, which shows CAQ compressed versions with three different thresholds, illustrates these effects. The same point is made in a different way by Fig. 10, in which mean squared error is plotted against threshold for the tank farm picture (Fig. 7). The deterioration for low and high values of A is reflected in this plot. The difficulty of using an objective criterion like mean squared error to judge a technique which attempts a subjective match to human vision is illustrated, however, by a comparison of Figures 11 and 12. Both of these have the same mean squared error (1.6%), but it is difficult to conclude that they are of the same quality.

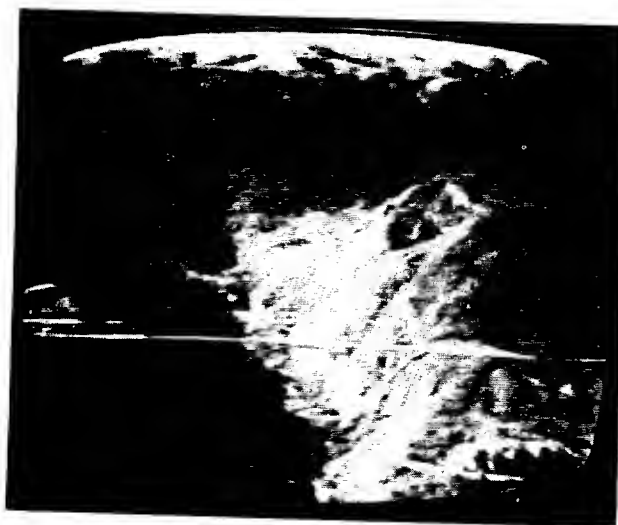


Fig. 7. Original tank farm image.

While the choice of A does not affect the amount of compression directly, it does vary the information content of the output by altering the proportion of P's, N's, and O's produced, and thus it affects the further saving that can be achieved by statistical coding. Fig. 13 indicates, for the same image, the average run length of zeroes as a function of threshold. Fig. 14 contains a plot of the entropy of the CAQ signal versus the average run length. Since the entropy places a lower limit on the number of bits per pixel that can be achieved without further loss, it is a good measure of the possible benefit of coding.

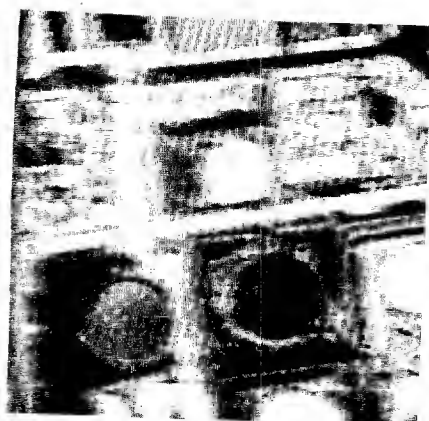


(a)

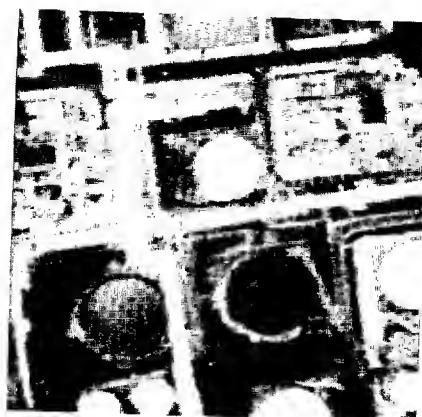


(b)

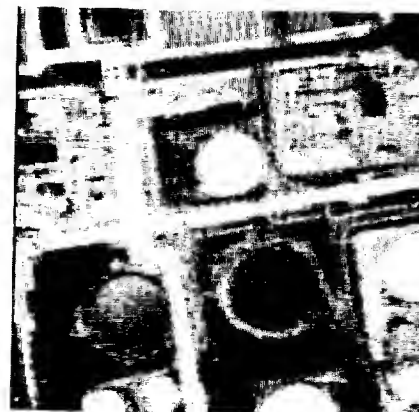
Fig. 8. Original (a) and compressed (b) imagery photographed from the monitor.



(a)



(b)



(c)

Fig. 9. CAQ - compressed images with three different thresholds: (a) $A = 2$, (b) $A = 12$, (c) $A = 32$.

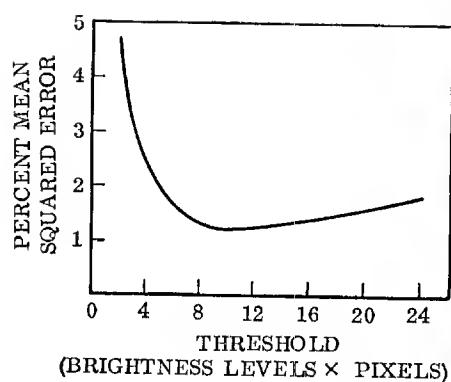


Fig. 10. Dependence of mean squared error on threshold for tank farm image.

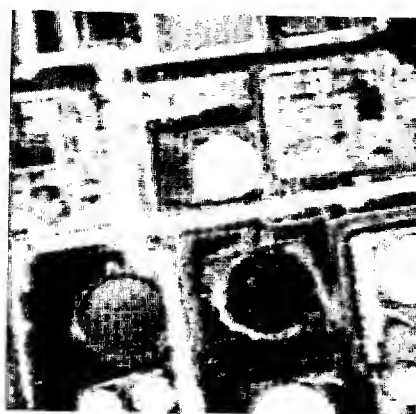


Fig. 11. CAQ - compressed image with $A = 6$.

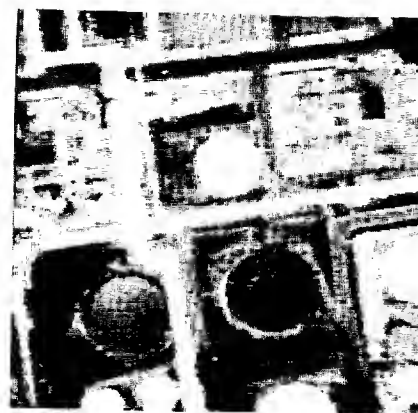


Fig. 12. CAQ - compressed image with $A = 20$.

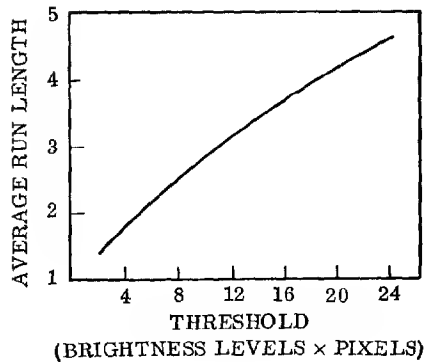


Fig. 13. Dependence of average run length on threshold for tank farm image.

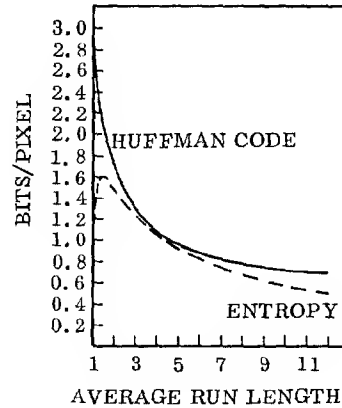


Fig. 14. Entropy of CAQ signal and Huffman-coded bits per pixel as a function of average run length for tank farm image.

The actual performance of the Huffman code is plotted on the same axes, and it can be seen to approach the entropy curve very closely in the region of highest picture quality. The probabilities given in Table I, which were used in determining the optimum code, were obtained by quantizing the tank farm scene, using a threshold of 16, and counting the P's and N's. P's, N's, and O's were assumed to be statistically independent and P's and N's to be equally probable. Different scenes, different thresholds, and more accurate statistics lead to slightly different codes, but the differences in performance among them are essentially negligible over normal operating ranges. The system is designed to pass one bit per pixel. Since the run length versus threshold curve (Fig. 13) is picture dependent, a tradeoff exists between picture degradation caused by too high a threshold, and line truncation produced by too short an average run length. This tradeoff can be made by adjusting the threshold. The best threshold for the busy image of Fig. 7 was found to be 16, while the optimum for the video scene (Fig. 8) was 8.

The picture quality is affected to some small degree by the reconstruction method chosen. Figures 15 and 16 compare the results using the "slope" and "step" reconstructions discussed earlier. Although the former gives a closer approximation to the original, in a mean squared error sense, the subjective appearance of the picture was not felt to be improved enough to justify the additional complexity.

Attempts to compare the CAQ technique with others must necessarily involve subjective judgements. Since it is a one-dimensional algorithm, it does not, of course, exploit correlations in the other direction. An 8x8 block Hadamard encoding with compression comparable to Fig. 16 is included for comparison as Fig. 17.

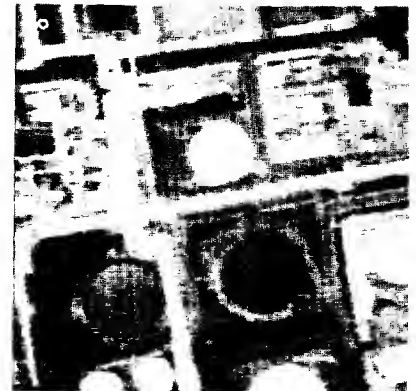


Fig. 15. "Slope" reconstruction of CAQ image (A = 16).

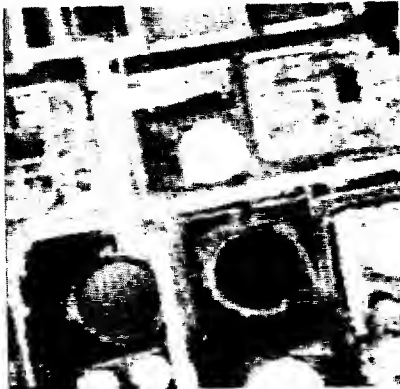


Fig. 16. "Step" reconstruction of CAQ image (A = 16).



Fig. 17. Two-dimensional Hadamard compressed image.

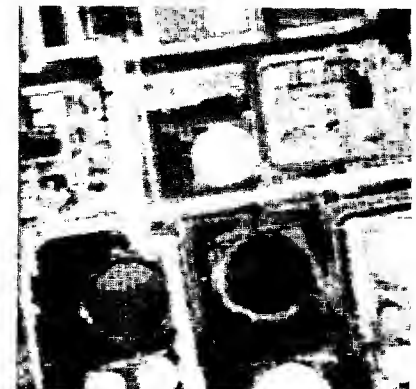


Fig. 18. Three level DPCM image.

The closest relatives of the CAQ are the class of predictive techniques, such as Differential Pulse Code Modulation (DPCM), and it is very similar to them in its response to bit errors. CAQ differs from these predictive techniques in quantizing the brightness-distance product, rather than merely the brightness. A picture produced by a three level DPCM is shown in Fig. 18 and can also be compared with the CAQ picture in Fig. 16. The DPCM levels are symmetric around zero change and the threshold was chosen to give the same entropy (1.1 bits per pixel) as the CAQ picture. The mean squared errors were 1.41% for the CAQ and 1.54% for the DPCM.

In summary, the CAQ technique provides a relatively simple approach, which can be implemented with conventional and available components to meet the size, weight, and power restrictions of RPV's. The technique requires no image storage for encoding; it exploits properties of human vision to provide pictures of good subjective quality using only one bit per pixel; and it has been implemented to operate over a full range of frame rates up to thirty frames per second.

Acknowledgements

This bandwidth compression system has been the work of a large number of people. The CAQ concept was originated a number of years ago by G. C. Sziklai. His advice and suggestions strongly influenced the development of the system, particularly in the early stages. Algorithm development, computer simulation and coding analysis were performed by C. D. Kuglin. The CAQ circuitry and the Huffman encoder and decoder were designed by M. Hilsenrath. The scan convertor and video circuitry were the work of B. S. Golosman and H. Massey.

INTRAFRAME AND INTERFRAME ADAPTIVE TRANSFORM CODING

Clifford Reader
Aeronutronic Ford Corporation
3939 Fabian Way, Palo Alto, California 94303

Abstract

This paper describes work performed as an extension of the techniques of transform coding with block quantization. These techniques have been very well defined theoretically with implementation of the predicted optimum results. The purpose of this work was to examine the techniques in practice and to define adaptive coding techniques which could be practically applied to the coding of still and moving pictures.

Introduction

The techniques of transform coding and block quantization for a single image are well established, the purpose of the work described in the first part of this paper was to examine in detail the application of these techniques in practice [1]. It is to be expected that there will be a variability of performance since any practical implementation will be rather removed from the theoretical optimum case with two dimensional Karhunen-Loeve sampling according to a known covariance and quantization based upon assumed statistics. In particular it is noted that the condition of stationarity of statistics will be violated and that for ease of implementation separable sub-optimum transformations - in this case the Slant - will be used over sub-blocks (16 x 16 pixels) of the image. A range of adaptive coding techniques is examined with practicality as well as coding efficiency being of importance.

The work is then continued into interframe coding. A coder is described which achieves a significant reduction in bit rate while not requiring the full frame memory of other interframe coders. The error performance of the coder is examined and static examples are presented. A discussion follows of the visual effect of the coder in real time together with the projected buffer requirements.

Intraframe Coding

Summary of Coding Procedure

The image is slant transformed in sub-blocks of 16 x 16 pixels. To perform quantization, the data is modelled as a Markov process and from this the variances of the transform domain samples are estimated. Coding bits are allocated in proportion to these variances and the samples are normalized in proportion to their estimated variances before being quantized according to the Max quantizer. Tuning of the process is possible by manipulation of the row and column correlations used to define the variances and with the constant of proportionality used in normalizing by the variances. The apparently arbitrary way in which tuning of the parameters influenced the coding error provided the motivation for this work.

Adaptive Coding 1 - The Amplitude Parameter

The constant of proportionality (C) in the normalization of the data is termed the amplitude parameter (AP). It was found that each sub-block of an image required a different AP for optimal coding (in mean squared error sense) and that although for most sub-blocks a good approximation could be made with one AP, in certain significant cases a markedly different - higher - AP was required suggesting that an adaptive process was required. This was effected by making the constant C a function of the square root of the energy of the sub-block less the D.C. component, i.e., the A.C. energy. This is termed the amplitude factor AF. Thus $C = AP \times AF$. The result was as follows. In non-adaptively coded images, the best result is obtained with an AP high enough to encode the problem sub-blocks and high frequency quantization noise then appears on the other sub-blocks. In the adaptive coding, the AF automatically compensates for the high energy problem sub-blocks such that one AP is more suitable for the whole image. This is seen as a freedom from noise in "flat" image areas and a slightly reduced mean square error. Figure 1.

Adaptive Coding 2 - Inter-Pixel Correlation

True horizontal and vertical correlations for natural images fall in the region of 0.91 to 0.96. Nevertheless, the effect was examined of using different correlations to define the model from which bit allocations and normalizing constants are derived. Results are summarized in Figure 2 where it can be seen that the lowest error is

encountered using far lower correlations than the real ones at the expense of increased sensitivity. What happens is that with the lower correlations, the available bits are spread further into the transform domain and more samples are encoded. Figure 3. However each sample receives fewer bits and thus the position of the quantization levels (as fine tuned by the AP) is more critical. Correlations of 0.76 were selected as a compromise with the highly satisfactory results shown in Figure 4.

Figure 1

Comparison of Adaptive and Non-Adaptive Coding
Correlations (0.86; 0.86) 1.5 bits per Pixel

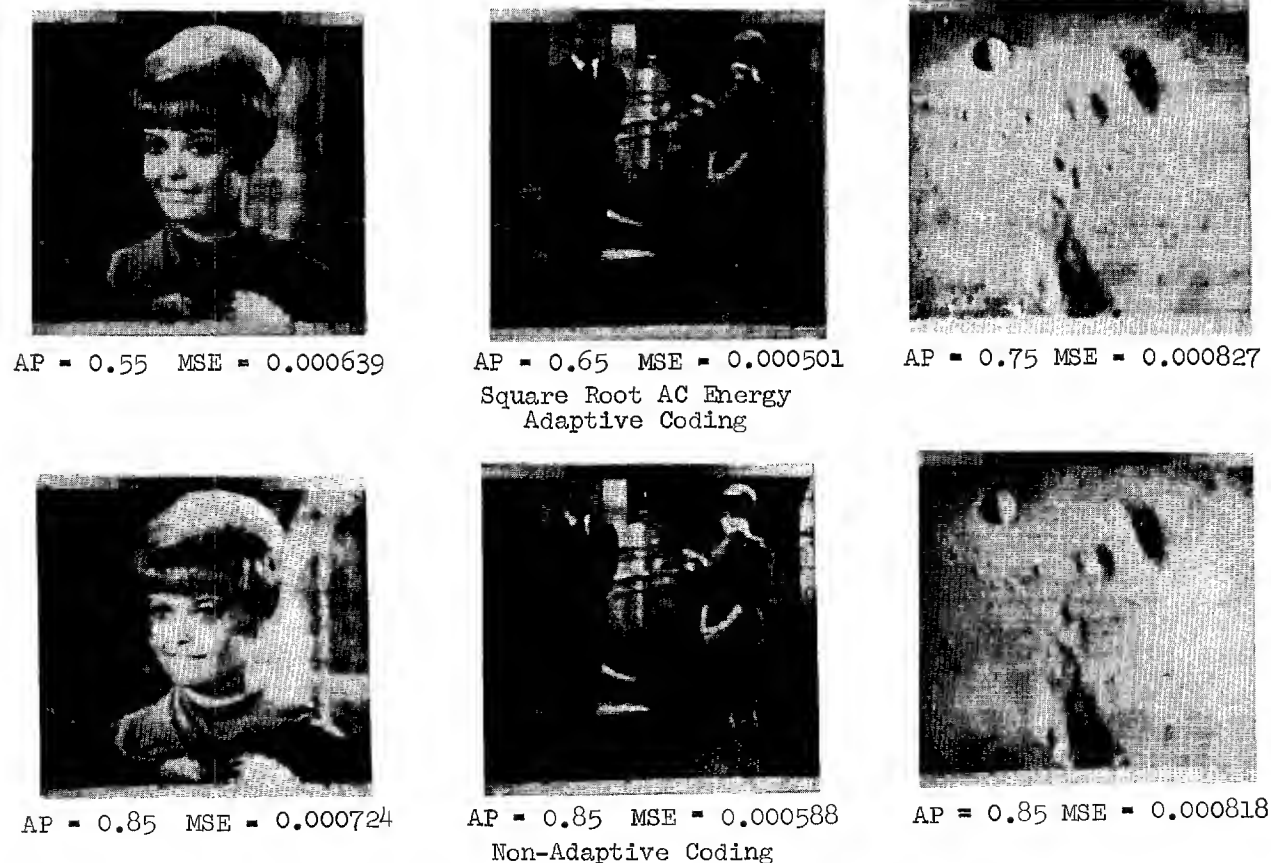


Figure 2

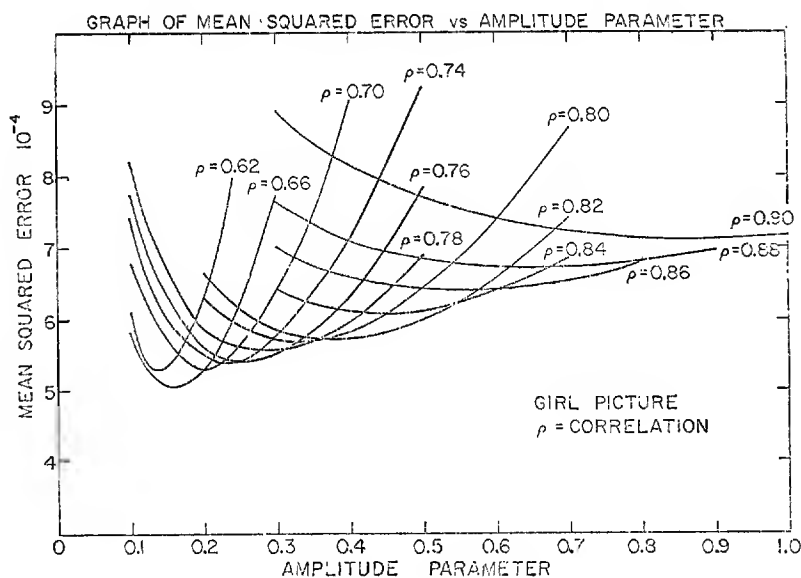


Figure 3
Bit Allocations

```

8 8 8 7 7 7 5 5 4 4 4 4 4 4 4
8 8 7 5 5 5 3 3 3 2 2 2 2 2 2
8 7 6 4 4 4 3 3 2 2 2 2 2 2 2
7 5 4 3 2 2 2 2 0 0 0 0 0 0 0
7 5 4 2 2 2 2 2 0 0 0 0 0 0 0
7 5 4 2 2 2 2 2 0 0 0 0 0 0 0
5 3 3 2 2 2 0 0 0 0 0 0 0 0 0
5 3 3 2 2 2 0 0 0 0 0 0 0 0 0
4 3 2 0 0 0 0 0 0 0 0 0 0 0 0
4 3 2 0 0 0 0 0 0 0 0 0 0 0 0
4 3 2 0 0 0 0 0 0 0 0 0 0 0 0
4 3 2 0 0 0 0 0 0 0 0 0 0 0 0
4 2 2 0 0 0 0 0 0 0 0 0 0 0 0
4 2 2 0 0 0 0 0 0 0 0 0 0 0 0
4 2 2 0 0 0 0 0 0 0 0 0 0 0 0
4 2 2 0 0 0 0 0 0 0 0 0 0 0 0

```

Correlations 0.86

```

8 8 8 6 5 5 4 4 4 3 3 3 3 3 3
8 7 6 5 4 4 3 3 3 3 3 2 2 2 2
8 6 5 4 4 4 3 3 2 2 2 2 2 2 2
6 5 4 3 3 3 2 2 2 2 2 0 0 0 0
5 4 4 3 2 2 2 2 2 2 2 0 0 0 0
5 4 4 3 2 2 2 2 2 0 0 0 0 0 0
4 3 2 2 2 0 0 0 0 0 0 0 0 0 0
4 3 3 2 2 2 0 0 0 0 0 0 0 0 0
4 3 2 2 2 0 0 0 0 0 0 0 0 0 0
3 3 2 2 2 0 0 0 0 0 0 0 0 0 0
3 3 2 2 2 0 0 0 0 0 0 0 0 0 0
3 2 2 0 0 0 0 0 0 0 0 0 0 0 0
3 2 2 0 0 0 0 0 0 0 0 0 0 0 0
3 2 2 0 0 0 0 0 0 0 0 0 0 0 0
3 2 2 0 0 0 0 0 0 0 0 0 0 0 0

```

Correlations 0.76

Figure 4
Coding with Square Root AC Energy Amplitude Factor
Correlations (0.76; 0.76)



AP = 0.25 MSE = 0.000544



AP = 0.30 MSE = 0.000446
1.5 Bits Per Pixel



AP = 0.35 MSE = 0.000749



AP = 0.25 MSE = 0.000863



AP = 0.3 MSE = 0.000757
1.0 Bits Per Pixel



AP = 0.35 MSE = 0.001023

Adaptive Coding 3 - Optimised Coding

The fine structure of the parameters required by each sub-block for optimum coding was examined by performing an experiment in which fifteen pairs of inter-pixel correlations (vertical and horizontal; Figure 5) were used to define bit allocations and normalizing constants which were applied in every combination (225) to every sub-block. Figure 6 shows sub-block maps in which the reference number (from Figure 5) is shown for the optimum combinations of bit allocations and normalizing constants. The resultant minimum mean squared error is shown in Figure 7 with the coded pictures shown in Figure 8. The pictures show good detail and edge rendition and it should be noted that even better coding could have resulted if correlations of less than 0.80 (type 1) had been allowed.

Figure 5

Correlation Reference No.	Correlations
1	0.80; 0.80
2	0.84; 0.84
3	0.88; 0.88
4	0.92; 0.92
5	0.96; 0.96
6	0.80; 0.92
7	0.80; 0.84
8	0.84; 0.92
9	0.88; 0.96
10	0.92; 0.96
11	0.92; 0.80
12	0.84; 0.80
13	0.92; 0.84
14	0.96; 0.88
15	0.96; 0.92

Figure 7

Sub-Block Map (Girl Image)
Optimum Mean Square Error

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	3	7	7	2	1	1	1	2	2
2	0	0	0	0	0	1	15	10	6	4	8	7	1	2	2	1
3	0	0	0	0	1	9	15	15	9	6	3	22	2	2	1	1
4	0	0	0	0	2	5	15	7	4	2	1	15	3	2	1	1
5	0	0	0	0	1	4	5	6	2	3	6	8	3	1	1	1
6	0	0	0	0	1	4	3	4	2	2	2	6	1	1	1	1
7	0	0	0	0	0	11	14	11	4	1	4	9	1	1	1	1
8	0	0	0	0	0	3	5	5	4	2	5	9	2	1	1	1
9	0	0	0	0	0	2	11	8	3	2	7	4	2	1	1	1
10	0	0	0	0	0	3	9	6	4	3	4	2	2	1	1	1
11	0	0	0	0	0	1	27	13	1	1	14	7	2	1	2	2
12	1	1	0	1	1	7	11	4	13	1	4	4	2	18	14	1
13	1	1	2	12	1	2	9	11	9	2	1	1	10	12	5	1
14	1	0	2	3	14	5	3	5	4	15	16	3	0	0	7	9
15	0	0	1	0	0	5	16	15	11	54	11	28	12	0	0	3
16	0	0	3	1	0	5	6	17	21	25	22	15	31	42	0	1

The Mean Square Error is coarsely approximated
1 = 0.0001

Figure 6

Sub-Block Maps (Girl Image)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	7	1	8	1	1	12	1	1	1
2	1	1	1	1	1	1	1	7	1	1	12	12	8	1	1	1
3	1	1	1	1	1	1	6	6	1	1	6	7	3	1	1	1
4	1	1	1	1	1	1	1	1	1	6	1	7	1	1	1	1
5	3	1	1	1	1	1	1	1	1	1	1	1	11	1	1	1
6	11	1	1	1	1	7	1	1	1	1	1	1	12	1	1	1
7	11	1	1	1	1	1	6	1	1	1	1	1	1	12	1	1
8	11	1	1	1	1	1	6	1	12	1	1	1	1	1	1	1
9	13	1	1	1	1	11	1	6	6	1	1	6	1	1	1	1
10	1	1	1	1	1	1	1	6	1	6	1	6	1	1	1	1
11	7	1	1	1	1	1	7	1	6	1	7	1	1	1	1	1
12	1	3	1	1	6	1	12	8	7	1	6	1	1	1	1	1
13	12	3	1	12	1	1	1	1	1	6	6	1	7	1	1	1
14	1	12	11	1	1	1	1	1	1	12	6	1	1	1	1	1
15	1	1	1	1	7	1	1	6	1	1	6	7	12	1	1	1
16	6	6	1	1	6	7	1	1	7	7	12	6	12	12	1	12

Sub-optimum Bit Allocations Types

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	10	10	10	1	1	3	1	1	1
2	1	1	1	1	1	1	5	1	1	6	3	6	15	1	1	1
3	1	1	1	1	1	1	1	6	8	8	6	6	4	1	1	1
4	1	1	1	1	1	1	6	1	1	1	1	3	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	13	1	1	1
6	1	1	1	1	1	2	1	1	1	1	1	1	2	1	1	1
7	1	1	1	1	1	1	8	2	2	1	1	8	1	3	1	1
8	1	1	1	1	1	5	6	3	6	1	1	1	1	1	1	1
9	1	1	1	1	1	15	1	6	8	1	1	1	1	1	1	1
10	1	1	1	1	1	1	6	6	1	1	1	1	1	1	1	1
11	6	1	1	1	1	1	6	6	1	1	7	1	1	1	1	1
12	4	10	1	1	9	6	1	11	5	8	3	8	8	1	1	1
13	10	4	1	3	1	1	1	1	1	1	6	1	13	1	1	1
14	5	11	1	1	1	1	1	1	1	1	8	1	1	1	1	1
15	1	1	1	1	2	1	1	2	1	4	8	10	2	1	1	1
16	7	1	1	1	7	6	1	1	9	3	3	8	5	5	1	15

(ii)

Sub-optimum companding Constant Types

Sub-blocks marked with a circle may be fairly well coded with bit allocations and companding constants of Type 1.

Adaptive Coding 4 - Zonal and Threshold Coding

The coding thus far described employs bit allocations in which those samples which are quantized fall inside a maximum variance zone. The coding error may be divided into the quantization error and the error resulting from those samples outside the zone which do not get coded at all. There will always be exceptional significant samples in the latter category and they could be coded by setting a magnitude threshold in the area outside the zone and run-length coding all samples exceeding such threshold. As an example, four bits were allocated to code each of such samples and, assuming that a line starting code were used, four bits were required to address each sample. The number of thresholded samples is shown by the sub-block maps of Figure 9 where it can be seen that a picture which has been zoned coded at 1.5 bits per pixel now requires 1.7 bits per pixel. In order to get back to 1.5 bits per pixel, the zonal coding

must be effected at 1.2 bits per pixel. The corresponding coded pictures are shown in Figure 10. It is considered that the result obtained at a total of 1.5 bits per pixel is not significantly improved and does not justify the work involved especially since channel error would be a problem.

Figure 8
Optimised Coding



Original



1.5 Bits Per Pixel
MSE = 0.000744



1.0 Bits Per Pixel
MSE = 0.000796

Figure 9

Sub-Block Maps (Girl Image)
The Number of Threshold Samples

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	5	15	7	1	0	0	0	1	1
2	0	0	0	0	0	0	20	19	14	6	11	11	0	1	3	0
3	0	0	0	0	0	10	19	33	25	12	4	21	0	0	0	0
4	0	0	0	0	2	8	21	8	6	2	0	17	3	2	0	1
5	0	0	0	0	0	5	11	11	0	3	14	10	4	0	0	0
6	0	0	0	0	0	7	1	1	0	2	1	10	0	0	0	1
7	0	0	0	0	0	14	23	18	4	1	2	13	0	0	0	0
8	0	0	0	0	0	2	7	8	7	0	6	10	1	0	0	0
9	0	0	0	0	0	1	10	9	3	2	9	0	1	0	0	0
10	0	0	0	0	0	3	11	9	6	4	0	0	0	0	0	0
11	0	0	0	0	0	0	29	26	0	0	23	7	0	0	0	0
12	0	0	0	0	22	23	9	20	0	5	3	3	23	31	0	1
13	0	0	2	21	1	0	12	17	11	1	1	0	18	21	6	0
14	0	0	0	6	25	9	1	2	8	22	30	0	0	0	2	20
15	0	0	0	0	0	6	30	22	21	45	19	41	19	0	0	2
16	0	0	2	0	0	7	13	22	28	29	39	20	55	37	0	0

(i)

1.7 bits per Pixel

Total Number of Threshold Samples = 6,008

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	7	21	14	8	0	0	0	1	4
2	0	0	0	0	0	1	36	26	18	14	23	20	0	1	3	2
3	0	0	0	0	0	18	28	47	34	18	10	43	2	0	0	0
4	0	0	0	0	2	14	31	20	11	5	1	26	6	2	0	1
5	0	0	0	0	1	12	18	21	8	6	22	23	7	0	1	0
6	0	0	0	0	0	16	4	6	6	5	2	18	0	0	1	1
7	0	0	0	0	0	27	35	35	18	4	9	25	0	0	0	0
8	0	0	0	0	0	7	9	14	14	2	15	15	1	0	0	0
9	0	0	0	0	0	2	24	21	6	3	20	0	1	0	0	0
10	0	0	0	0	0	7	18	18	8	5	0	1	1	0	0	0
11	0	0	0	0	0	4	39	39	0	0	39	18	0	1	0	2
12	1	0	0	0	32	34	12	33	0	11	8	6	43	43	0	2
13	0	0	5	38	1	0	23	29	27	5	2	8	32	33	16	0
14	1	0	0	11	40	19	4	13	17	40	47	4	0	0	13	31
15	0	0	3	0	0	20	44	34	36	70	33	59	36	0	0	5
16	0	0	10	1	0	20	37	48	47	55	29	75	60	0	0	0

(ii)

1.5 bits per Pixel

Total number of Threshold Samples = 10,532

Figure 10
Threshold Coding



Original



1.7 Bits MSE = 0.000533

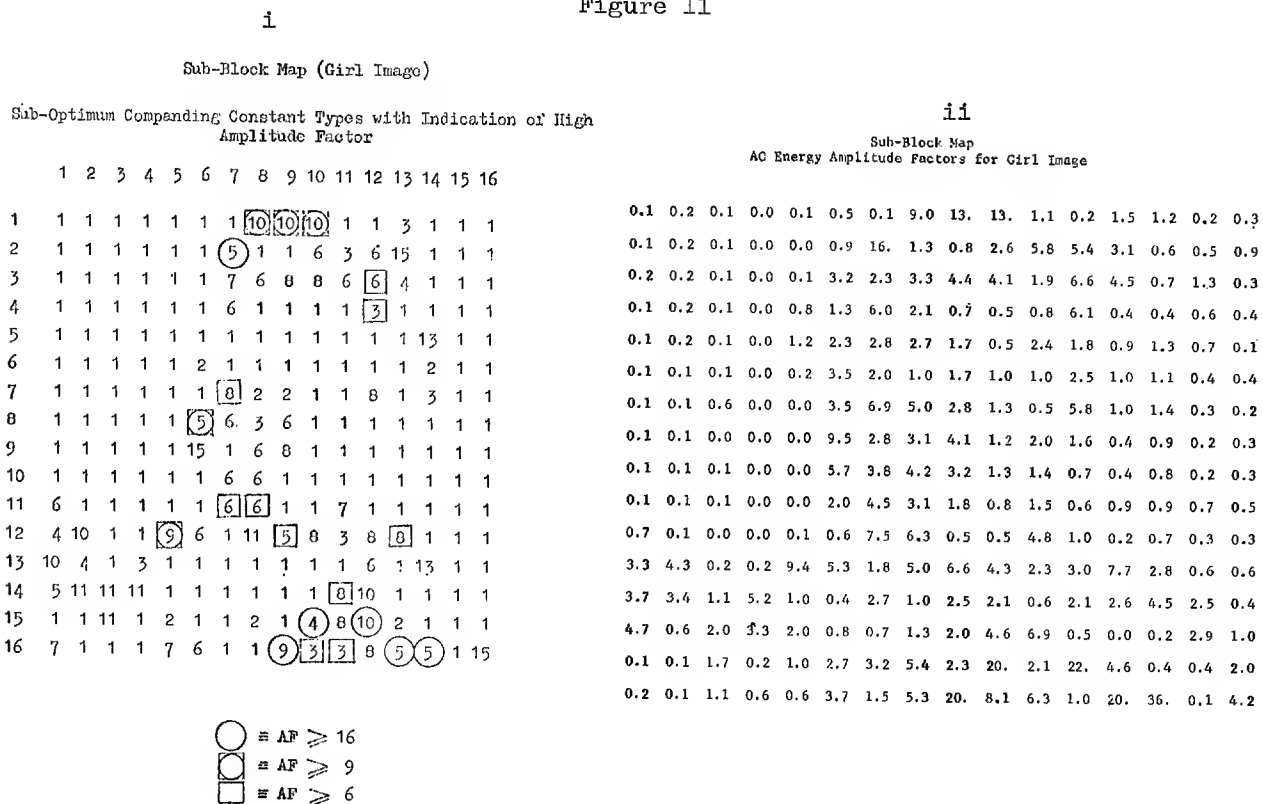


1.5 Bits MSE = 0.000606

Adaptive Coding 5 - Zonal Coding

If the sub-block maps of Figures 6, 7 and 9 are studied, then it is seen that there is high correlation between them in terms of which sub-blocks display exceptional information. Two further maps are presented in Figure 11.

Figure 11



Now it can be seen that certain sub-blocks require exceptional bit allocations and normalizing constants, have many significant samples without a normal bit allocation zone, produce a high mean squared error even when optimally coded and furthermore possess the highest AC energy. This last property allows easy identification of these sub-blocks and hence definition of an adaptive, variable zonal coding is possible. An example is presented to indicate the performance of such a scheme. The most difficult sub-blocks will require many bits in order to be coded satisfactorily, e.g. 3 bits per pixel then the relatively high energy sub-blocks might be coded well at 2 bits per pixel. The effect for these images is shown in Figure 12 where the bit rate for the remaining (background information) sub-blocks is shown calculated to yield an overall 1.5 bits per pixel.

Figure 12

Image	Number of sub-blocks at 3.0 Bits per pixel	Number of sub-blocks at 2.0 Bits per pixel	Number of sub-blocks remaining	Rate of remaining sub-blocks Bits per pixel
Girl	11	127	118	0.82
Couple	3	120	133	1.02
Moon	4	71	181	1.27

The distribution of bits is shown in Figure 13 and it is immediately obvious that the AP is a very effective delineator of detail in the picture. Excellent quality would result from such a scheme with the only problem occurring when the discreteness of the sub-blocks causes part of a feature to be missed.

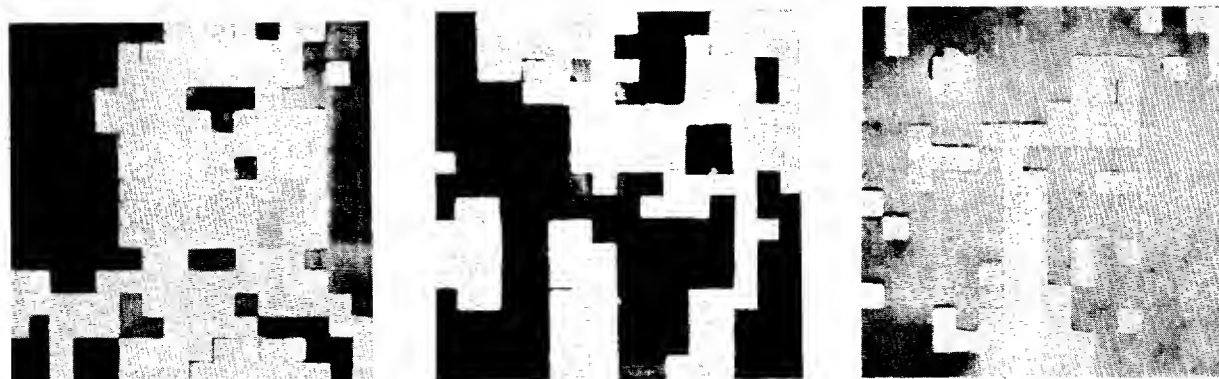
Conclusion - The Application of Statistical Coding

The results presented in previous sections show that most areas of the images are coded effectively by straightforward application of the statistical models. The exceptions, when they occur are significant because they involve prominent features.

Figure 13
Adaptive Zonal Coding



Sub-Blocks Coded at 3 Bits Per Pixel



Sub-Blocks Coded at 3 Bits Per Pixel Plus Those Coded at 2 Bits Per Pixel

Typically these are characterized by a sharp high contrast edge. Obviously, in a sub-block containing such features the statistics are not Markovian and not stationary. No manipulation of the maximum variance zone except to simply increase its size is effective in coding this data. As an example, the transform domain of the sub-block located on the diagonal edge of the girl's hairline (reference Figure 13) is shown in Figure 14.

Figure 14

Slant Transform domain of Sub-block (2,7)

Girl Image

6.080	<u>-2.038</u>	<u>-0.173</u>	<u>-0.051</u>	0.056	<u>-0.099</u>	0.005	0.002	<u>-0.014</u>	<u>-0.008</u>	0.025	0.002	0.008	<u>-0.040</u>	<u>-0.001</u>	<u>-0.039</u>
<u>-2.951</u>	<u>-0.284</u>	<u>0.914</u>	0.006	<u>0.210</u>	<u>-0.040</u>	<u>-0.013</u>	<u>-0.016</u>	<u>-0.001</u>	0.006	<u>-0.021</u>	<u>-0.004</u>	0.011	0.024	0.027	0.015
<u>0.185</u>	<u>1.060</u>	<u>-0.049</u>	<u>-0.389</u>	<u>-0.009</u>	<u>-0.258</u>	0.004	0.078	0.062	<u>-0.093</u>	0.019	0.039	<u>-0.035</u>	<u>-0.014</u>	<u>-0.009</u>	<u>-0.038</u>
<u>0.135</u>	0.001	<u>-0.394</u>	<u>-0.055</u>	<u>0.161</u>	<u>-0.026</u>	0.008	<u>-0.072</u>	<u>-0.007</u>	<u>-0.011</u>	<u>-0.007</u>	<u>-0.006</u>	<u>-0.009</u>	0.000	0.000	0.049
<u>-0.269</u>	<u>0.418</u>	<u>-0.042</u>	<u>0.229</u>	<u>-0.016</u>	0.072	0.070	<u>-0.121</u>	0.029	0.001	0.028	<u>-0.124</u>	0.003	<u>-0.007</u>	<u>-0.013</u>	<u>-0.012</u>
<u>0.108</u>	0.055	<u>-0.543</u>	0.090	<u>-0.003</u>	0.003	<u>0.119</u>	<u>-0.064</u>	<u>-0.043</u>	0.034	<u>0.109</u>	<u>-0.022</u>	0.012	<u>-0.017</u>	<u>-0.018</u>	0.031
<u>0.123</u>	<u>-0.140</u>	<u>-0.049</u>	<u>0.144</u>	<u>-0.106</u>	<u>0.231</u>	<u>-0.070</u>	0.058	0.065	<u>-0.018</u>	<u>-0.001</u>	0.020	0.001	<u>-0.001</u>	0.016	<u>-0.008</u>
0.037	<u>-0.007</u>	<u>0.038</u>	<u>-0.021</u>	<u>-0.226</u>	0.068	<u>-0.045</u>	0.074	<u>-0.009</u>	<u>-0.057</u>	<u>-0.013</u>	0.014	<u>-0.005</u>	0.007	0.015	<u>-0.021</u>
<u>-0.018</u>	0.029	0.084	<u>-0.030</u>	0.054	<u>-0.053</u>	<u>-0.059</u>	0.059	<u>-0.037</u>	0.020	<u>-0.026</u>	0.054	0.004	0.001	0.034	<u>-0.001</u>
0.032	<u>-0.065</u>	<u>-0.047</u>	<u>-0.024</u>	0.013	<u>-0.020</u>	<u>-0.039</u>	0.053	0.000	0.033	<u>-0.061</u>	0.036	<u>-0.022</u>	0.032	<u>-0.026</u>	0.011
0.099	<u>-0.089</u>	0.074	0.053	<u>-0.030</u>	0.063	0.012	<u>-0.016</u>	<u>-0.088</u>	<u>-0.002</u>	<u>-0.005</u>	0.002	0.020	0.031	<u>-0.016</u>	<u>-0.001</u>
0.015	<u>-0.082</u>	0.011	<u>-0.046</u>	<u>-0.062</u>	0.024	<u>-0.003</u>	0.027	0.026	0.041	0.002	0.014	0.003	<u>-0.033</u>	0.016	<u>-0.026</u>
<u>-0.006</u>	<u>-0.095</u>	<u>-0.030</u>	<u>-0.001</u>	0.026	0.020	0.060	0.003	0.009	<u>-0.006</u>	0.038	<u>-0.016</u>	0.022	<u>-0.015</u>	<u>-0.020</u>	<u>-0.012</u>
0.007	0.056	0.079	<u>-0.023</u>	<u>-0.021</u>	<u>-0.044</u>	<u>-0.029</u>	<u>-0.030</u>	0.019	<u>-0.029</u>	<u>-0.005</u>	<u>-0.001</u>	<u>-0.002</u>	0.003	0.017	0.019
0.001	0.028	<u>-0.056</u>	<u>-0.007</u>	<u>-0.009</u>	0.005	<u>-0.001</u>	<u>-0.022</u>	0.012	0.012	<u>-0.007</u>	<u>-0.053</u>	0.002	0.032	0.021	<u>-0.029</u>
<u>-0.046</u>	0.028	0.014	<u>-0.010</u>	0.037	<u>-0.012</u>	0.034	<u>-0.017</u>	0.020	<u>-0.020</u>	0.026	0.015	0.012	<u>-0.035</u>	<u>-0.014</u>	0.015

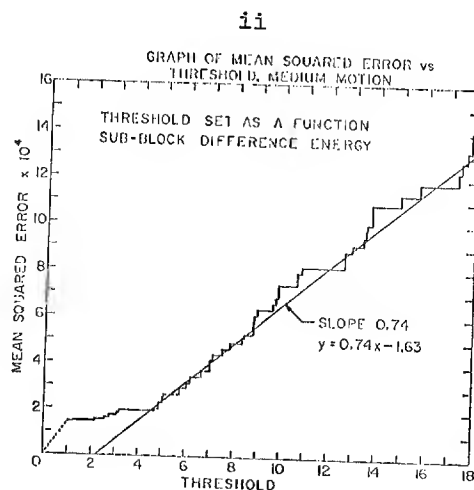
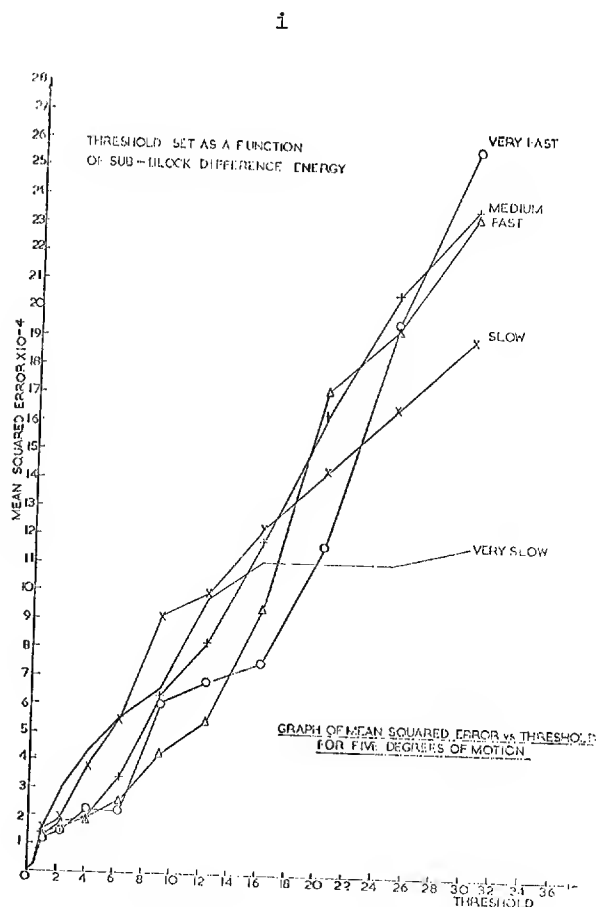
The significant samples have a distribution out along the leading diagonal, very unlike the maximum variance zone. In consequence, the adaptive zonal coding is recommended as best, particularly since bits are concentrated where desired - in the girl image, the girl herself is coded mostly with two bits and the insignificant background with only 0.82 bits. The moon picture, however, has the few outstanding features coded at 2 or 3 bits but the more noticeable background receives 1.27 bits.

Lastly it is noted that the mathematical model approach to coding has now been abandoned in favor of zones based upon measured statistics (variances) an improved coding results since the requirements of the problem sub-blocks are built-in.

Interframe Coding

It is well known that considerable redundancy is present when a moving picture is transmitted. The adaptive coder described here was designed to take advantage of the limitations of human vision, essentially providing only sufficient information to yield the illusion of a complete high detail moving picture. The coder is specifically designed to minimize terminal cost in addition to minimizing transmission cost. In addition a two stage coding is applied in order to obtain flexibility in implementation. An application was picture telephone in which simple coding might be used locally with additional coding for long distance. The first stage of coding consists of the removal of spatial redundancy within single frames. This employs the techniques discussed earlier. The second stage of coding - interframe coding is thus presented with coded transform domain samples at a rate considerably less than the source rate. The bit rate reduction is to be achieved by the updating of only those portions of the frame in which movement has taken place since the previous frame. Since the interframe coder is presented with data which has been transformed in sub-blocks, it is necessary to decide whether a particular sub-block contains movement and if so, to update it. Detection of the sub-blocks which contain motion may be effected by examination of the sub-block frame difference energy with a threshold decision of whether to transmit a new sub-block or not. The effect is a remarkably linear one as indicated by the error curves of Figure 15.

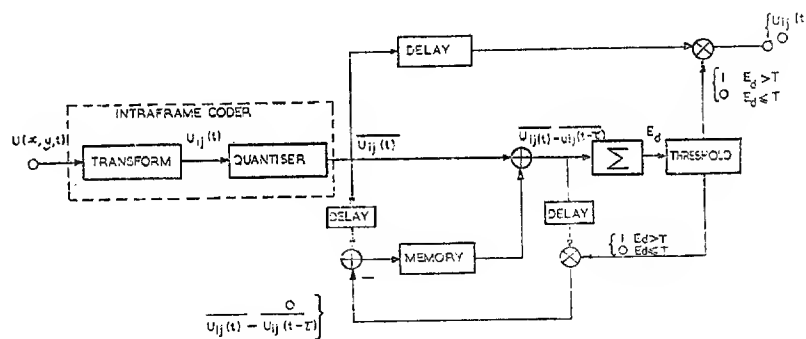
Figure 15



Serious deviation occurs only for very slow motion where the discreteness of the update process becomes apparent. The block diagram for an implementation of the system is shown in Figure 16. Shown is the difference energy summing and thresholding, the memory and delays which compensate for the time to make the update decision. In a normal coder the memory would have to store a complete frame. The advantage of this coder is that compression has already been achieved by the intra-frame coder and the same degree of compression applies to the size of the memory. A 4:1 compression (6 bits to 1.5 bits) is likely and this results in only a one quarter frame memory. From the curve of Figure 15 (ii) a threshold of 4 is selected as offering the best compromise between updating all

moving areas but not updating differences due to noise. The result is shown in Figure 17. Some blur is visible from the intraframe coding but the interframe coding is perfect. The result with a threshold of 9 is shown in Figure 18. Some updating errors are visible.

Figure 16



BLOCK DIAGRAM OF ADAPTIVE INTERFRAME CODER

Owing to the non-availability of data, it was not possible to test the coder dynamically - possibly a problem might be visibility of the sub-block updating structure. This could be coped with by overlapping sub-blocks or careful transmission of the DC and low frequency information. The buffering problems associated with the variable data rate produced by the coder were found to be similar to those encountered with a pixel cluster coding update scheme. In particular, in the case of buffer overload it is noted that an elegant reduction of data input may be effected by a further spatial compression (re-quantization or removal of transform domain samples) to take advantage of the limited spatial resolution of the eye to moving images.

Figure 17
Interframe Coding with Threshold of 4.0
Four Frame Sequence



(i) Intraframe coding at 3.0 bits per pixel

Figure 17 (Continued)

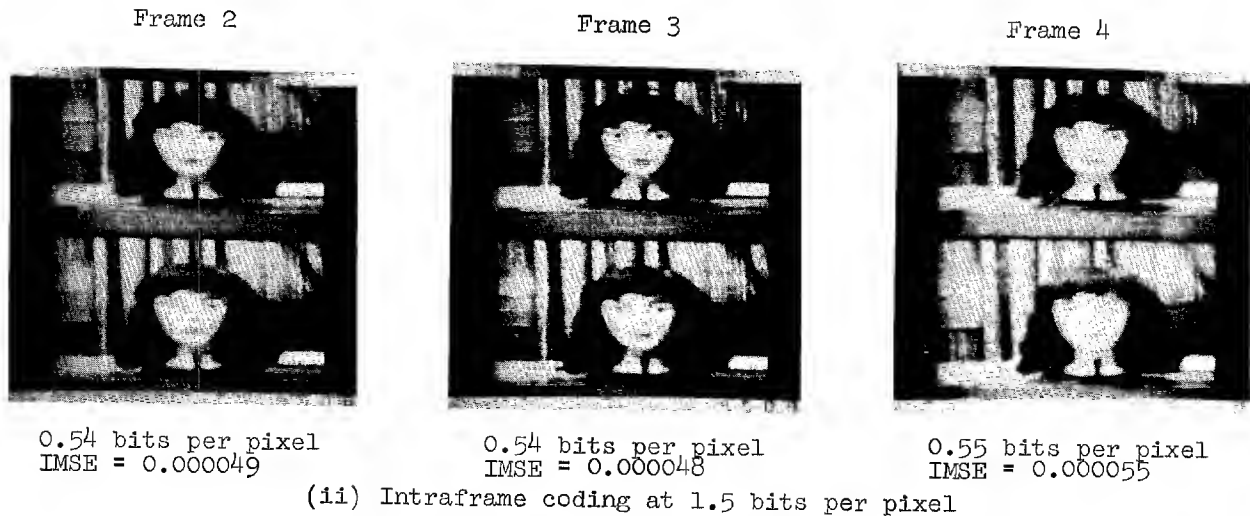
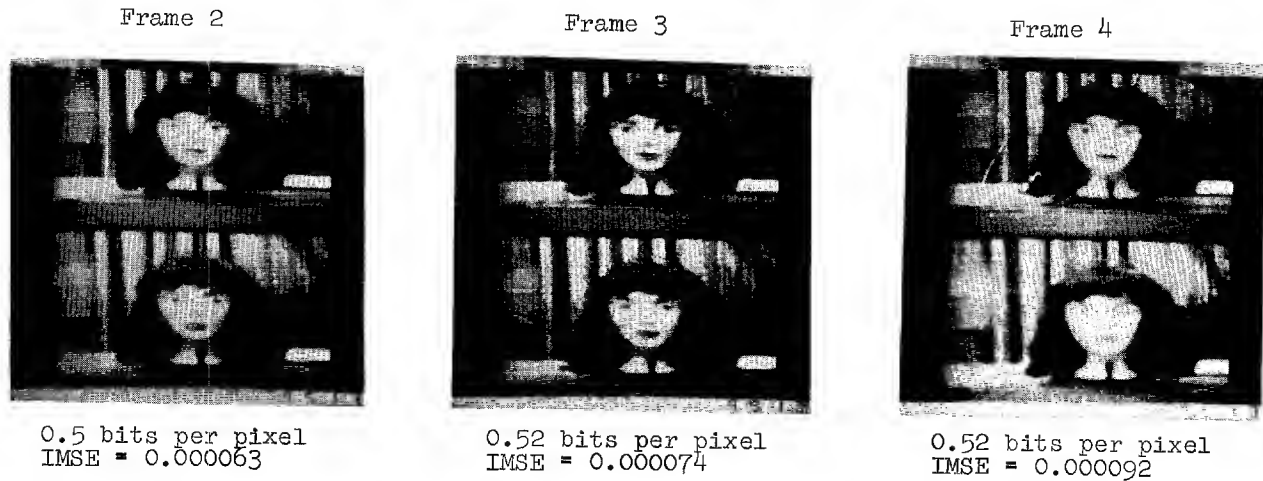


Figure 18
Interframe Coding with Threshold of 9.0
Four Frame Sequence
Intraframe Coding at 1.5 bits per pixel



References

1. "Orthogonal Transform Coding of Still and Moving Pictures", C. Reader, Ph.D. Thesis, Sussex University, England, 1974.

Session III
IMPACTS OF IMAGE DATA COMPRESSION

Panel Moderator
ANDREW G. TESCHER
The Aerospace Corporation

Session IV
POTENTIAL IMPLEMENTATIONS

Chairman
J. R. PARSONS
The Aerospace Corporation

JOINT PATTERN RECOGNITION/DATA COMPRESSION CONCEPT
FOR ERTS MULTISPECTRAL IMAGING*

Edward E. Hilbert
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

Abstract

This paper describes a new technique which jointly applies clustering and source encoding concepts to obtain data compression. The cluster compression technique basically uses clustering to extract features from the measurement data set which are used to describe characteristics of the entire data set. In addition, the features may be used to approximate each individual measurement vector by forming a sequence of scalar numbers which define each measurement vector in terms of the cluster features. This sequence, called the feature map, is then efficiently represented by using source encoding concepts. A description of a practical cluster compression algorithm is given and experimental results are presented to show trade-offs and characteristics of various implementations. Examples are provided which demonstrate the application of cluster compression to multispectral image data of the Earth Resources Technology Satellite.

Introduction

This paper further describes a new technique which jointly applies clustering and source encoding concepts to obtain data compression. A practical algorithm for implementing this concept was initially discussed in a previous paper [1]. This cluster compression technique is applicable to multidimensional information sources in general, but particular emphasis is given here to compressing multispectral image data from future operational Earth Resources Technology Satellites (ERTS). A Joint Pattern Recognition/Data Compression Model is defined to illustrate the general concept of combining pattern recognition and data compression. Then a practical realization of the cluster compression concept is described and experimental results are presented which show trade-offs and characteristics of various implementations. Examples are also provided which demonstrate the application of cluster compression to multispectral image data from ERTS.

Data Model

The multispectral image source can be modeled by the continuous random process $s(y_1, y_2, w)$ which is the electromagnetic energy at wavelength w for spatial coordinates y_1 and y_2 . The measurement vector elements of a digitized d-band multispectral image are then represented by the vector $\underline{X}(y_1, y_2)$ obtained from $s(y_1, y_2, w)$ by discretizing the variables y_1 , y_2 and w to give

$$\underline{X}(y_1, y_2) = [s(y_1, y_2, w_1), s(y_1, y_2, w_2), \dots, s(y_1, y_2, w_d)]. \quad (1)$$

*This paper presents the results of one phase of research carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract No. NAS7-100, sponsored by the National Aeronautics and Space Administration.

A Joint Pattern Recognition/Data Compression Model

Figure 1 describes a model of the generalized concept for the cluster compression technique. Let $\{X_i\}_{i=1}^n$ be a sequence of n measurement vectors obtained from a subset of the multispectral image data. The entire sequence $\{X_i\}_{i=1}^n$ is analyzed to extract features $\{\psi_j\}_{j=1}^m$ for the sequence $\{X_i\}_{i=1}^n$. These features could also be considered as primitives, or as a basis for approximating measurement vectors. The sequence of m features $\{\psi_j\}_{j=1}^m$ provides a description of the characteristics pertaining jointly to the entire sequence of measurement vectors $\{X_i\}_{i=1}^n$ and in some applications may be the only output required. In other applications each measurement vector needs to be approximated in terms of $\{\psi_j\}_{j=1}^m$ by the sequence of scalar numbers $\{\gamma_k\}_{k=1}^p$ which assigns to each measurement vector an approximation in terms of one of the primitives. Alternately, the concept of Fuzzy Set Theory [2] might be used to allow each measurement vector to be described by a weighted mixture of more than one of the features in the sequence $\{\psi_j\}_{j=1}^m$. The scalar sequence $\{\gamma_k\}_{k=1}^p$ constitutes a spatial map of the measurement vectors in terms of the primitives, and is called the feature map. Spatial features can be extracted from $\{\gamma_k\}_{k=1}^p$, and source encoding can be used to efficiently represent the spatial characteristics of $\{X_i\}_{i=1}^n$ through the encoded feature map denoted by C_γ . For each sequence $\{X_i\}_{i=1}^n$ of measurement vectors the model basically uses pattern recognition concepts to determine a set of features or primitives, and then the model uses data compression techniques to efficiently represent the spatial features in terms of the primitives.

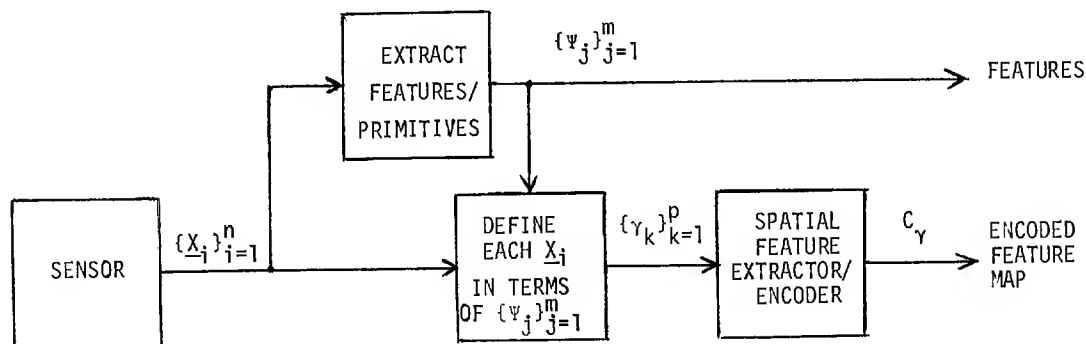


Fig. 1. A Joint Pattern Recognition/Data Compression Model.

The remainder of this paper describes a new practical method of data compression, called cluster compression, which is based upon the model in Fig. 1. Cluster compression uses clustering to extract multidimensional primitives from subsets of spatially contiguous measurements in the image data and then applies entropy encoding to the feature map. Another independently developed data compression technique related to the model in Fig. 1 is the Blob algorithm [3]. This algorithm is basically a boundary finding algorithm that guarantees closed boundaries. A blob is defined as a connected set of picture elements all of which have some common characteristic. In the Blob algorithm the primitives are defined by the boundary finding algorithm and consist of statistical descriptions of the measurement vectors contained within the various blobs. Spatial definition can then be provided by encoding a sequence which defines the spatial boundaries of the blobs. This algorithm is most useful as a data compressor when the image consists of well defined boundaries enclosing picture elements of nearly uniform characteristics, (e.g., agricultural fields). The Cluster Compression algorithm is different in that it extracts primitives and obtains significant compression independent of the spatial characteristics of the data. Thus the cluster compression technique can efficiently compress images whether or not they contain uniform areas with well defined boundaries. In addition, if the image does consist of uniform areas with well defined boundaries, such as fields, the entropy encoding of the feature map also results in efficient representation of these spatial boundaries.

CLUSTERING FOR EXTRACTION OF FEATURES

Let $\{X_i\}_{i=1}^n$ be a sequence of n d -dimensional spectral intensity measurements from a spatially local region of a multispectral image. These vectors generally tend to form groups in multispectral intensity space. A typical plot of $\{X_i\}_{i=1}^n$ for two bands in spectral intensity space is shown in Fig. 2(a). An intuitively natural set of features to use for representing $\{X_i\}_{i=1}^n$ in Fig. 2(a) are the descriptions for groups of measurement vectors, for example, the groups depicted in Figs. 2(b)-2(d). Each feature then consists of whatever set of parameters are used to describe the source distribution for the group of measurement vectors. The mean and covariance are an obvious example of a group description, or feature. The number of features extracted from any sequence of measurement vectors would depend on how accurately $\{X_i\}_{i=1}^n$ must be represented, and correspondingly what data rate is acceptable. An example of how two, four, or eight features might be chosen is shown in Figs. 2(b)-2(d).

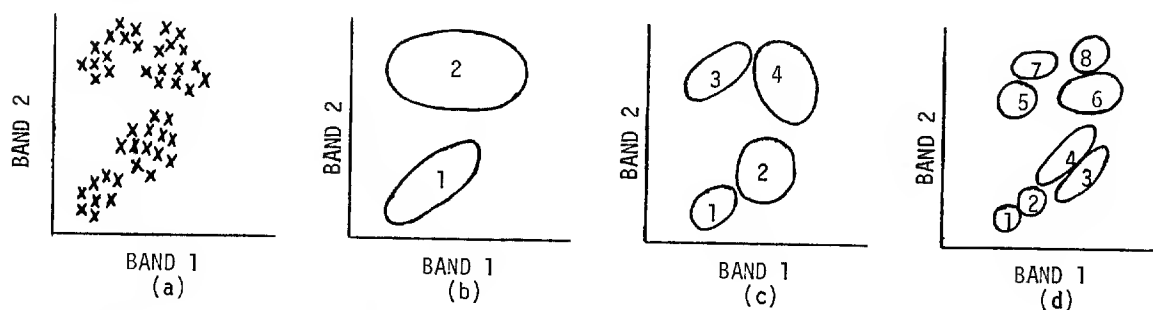


Fig. 2. Typical display in two band spectral intensity space for measurement vectors and for groups of measurement vectors used as features. (a) Measurement vectors $\{X_i\}_{i=1}^n$. (b) Two groups. (c) Four groups. (d) Eight groups.

Clustering Approaches

Clustering is a means for automatically grouping multidimensional data [4]-[6]. Figure 3 shows the basic iterative approach to clustering. All clustering techniques depend on a distance measure between vectors and a distance measure between clusters, or groups of vectors. The distance measure between vectors is usually either the Euclidean distance or the absolute value distance. A distance measure between clusters could be as simple as the distance between means, or more generally the shape of the two clusters could also be used in computing a measure of distance.

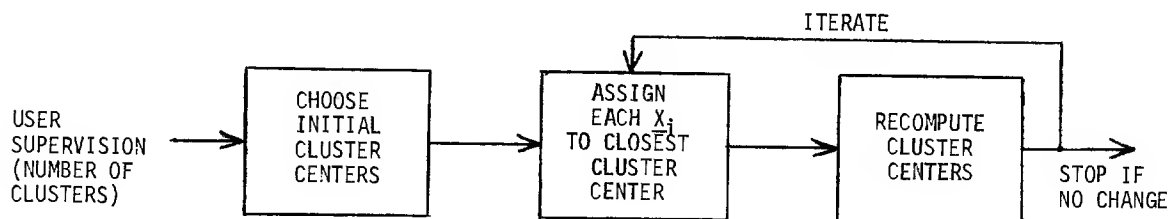


Fig. 3. The basic iterative approach to clustering

The simple clustering algorithm of Fig. 3 is defined as the Basic Clustering Algorithm (BCA), and simulation results involving its use in cluster compression are presented later in this paper. The repetitive assignment computations for the BCA are very simple, and the algorithm can be structured in a highly parallel manner making very high data rates feasible. The only supervision required in this clustering is the choice of how many clusters are to be obtained for each set of measurement vectors. Even when the number of clusters for each data set is constant, the self scaling characteristics of clustering provide

a good multidimensional quantization of spectral intensity space. However, for some applications it is desirable to have the number of clusters used per data set be variable and adaptively determined to meet certain quality requirements. There are many ways of making the basic iterative clustering algorithm in Fig. 3 adaptive. For example, clusters might be split, combined, or deleted based on intracluster and intercluster distance measures. A widely used algorithm, ISODATA, with these adaptive traits was originated by Ball and Hall [7]. Modifications of ISODATA for ground data evaluations of ERTS multispectral data produced ISOCLS [8], [9]. The Adaptive Clustering Algorithm (ACA) referred to later in the simulation examples is basically ISOCLS with some simplifying modifications. Another clustering approach often used with ERTS multispectral data is defined in [10]. This approach is similar to the ISOCLS approach except that a Swain-Fu distance is used as a more accurate measure of intercluster distance. However, the Swain-Fu distance also requires more computation. Any clustering approach could be used with the cluster compression concept, but the emphasis in this paper is on clustering approaches amenable to high data rate implementations. Furthermore, initial simulation results suggest that more sophisticated clustering is unnecessary.

Cluster Features

The features extracted from $\{X_i\}_{i=1}^n$ by clustering are typically a subset of the following: cluster means, cluster variances per band, cluster covariances, number of vectors in each cluster, set of intercluster distances, etc. Any description of a cluster can be considered a feature, even the set of vectors themselves. These types of cluster parameters comprise a compact description of a group of measurement vectors in a statistical format which is both easily interpreted and directly meaningful for image approximation and computer classification applications. An example of a simple cluster feature is the cluster mean.

Cluster Feature Rate. Let $\{\psi\}_{j=1}^m$ be the set of m cluster features which describe the sequence of n measurement vectors. Assume each ψ_j requires b bits of quantization per band. For example, in image approximation ψ_{j*} might be the j *th cluster mean with b equal to 6 (or 8) bits per band, or in classification uses ψ_{j*} might be the j *th cluster mean and variance with b equal to 9 (or 11) bits per band if the variance per band is defined to 3 bit resolution. The spectral rate R_{spec} is defined as the bits per picture element per band (bpppb) needed to define the spectral characteristics of $\{X_i\}_{i=1}^n$.

$$R_{\text{spec}} = \frac{m \cdot b}{n} \quad (2)$$

A graph of R_{spec} vs. m for various values of n is shown in Fig. 4 with b assumed equal to six, which corresponds to using only cluster means as features. In some cases this description of $\{X_i\}_{i=1}^n$ may be all that is needed, and R_{spec} then represents the total rate in bpppb.

FEATURE MAP SPATIAL DEFINITION

Feature Map Rate

In many applications it is desired to further approximate $\{X_i\}_{i=1}^n$ by approximating each vector individually in terms of the cluster features. For example, if vector X_{i*} belongs to cluster j *, then X_{i*} could be approximated by ψ_{j*} , which might be simply the mean of cluster j *. Alternately, X_{i*} could be approximated by a combination of $\{\psi_j\}_{j=1}^m$. The feature map sequence $\{\gamma_k\}_{k=1}^p$ defines the approximation

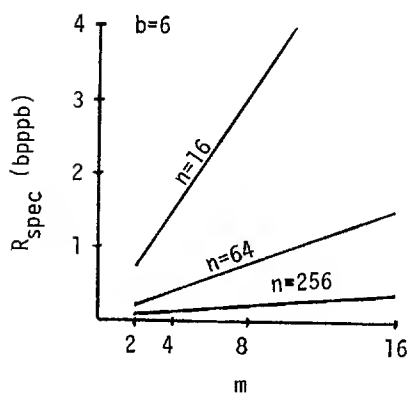


Fig. 4. Graph of spectral rate per band vs. number of features for various measurement vector sequence lengths with 6 bits per band per feature.

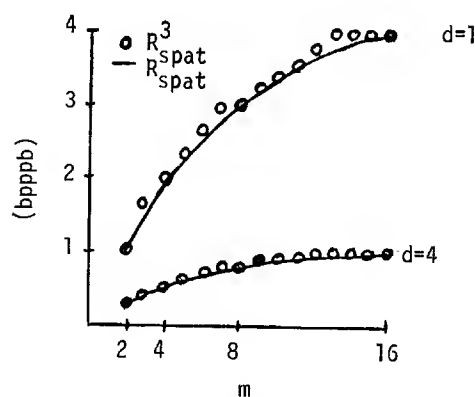


Fig. 5. Graph of spatial rate and practical spatial rate per band vs. number of features for 1 and 4 dimensional measurement vectors.

of each measurement vector and thus gives spatial definition for the sequence $\{X_i\}_{i=1}^n$. The data rate in bpppb for $\{Y_k\}_{k=1}^p$ is defined as R_{spat} . If each d -dimensional measurement vector is approximated by one of m primitives, then

$$R_{\text{spat}} = \frac{\log_2 m}{d} . \quad (3)$$

Practical System Rate. In practical systems, however, if m is not a power of two, the rate in (3) can only be approximated by representing extensions of the source. Let R_{spat}^3 be the rate achieved in practice when groups of three of the original m alphabet source are represented by natural coding. This representation of the source is easy to implement since m is small. Figure 5 shows R_{spat} and R_{spat}^3 vs. m for two values of d and shows R_{spat}^3 is never significantly greater than R_{spat} for any value of m which is of interest.

Feature Map Source Coding

Entropy Coding. A typical feature map sequence for an $\{X_i\}_{i=1}^n$ described by four cluster features is shown in Fig. 6(a). In image data there is significant spectral correlation between spatially close picture elements, and correspondingly the differences between spatially adjacent elements in the feature map have a sample distribution function similar to that shown in Fig. 6(b). The entropy for the distribution of differences in the feature maps is on the average lower than $\log_2 m$. Thus entropy coding techniques offer an opportunity to reduce R_{spat} without any degradation in the reconstructed feature map. The entropy for the distribution of differences for each $\{Y_k\}_{k=1}^p$ is often used as a measure of performance for the encoding of the differences. This entropy represents a performance bound for the encoding of a zero memory sequence with that sample distribution. Similarly, averaging the entropies over any number of sequences can be used as a measure of average performance for encoding sequences of differences. The average entropy represents an unachievable bound on the average performance for encoding all such sequences with zero memory, since it ignores the overhead required to identify the optimum code assignment for each sequence of differences.

Entropy Coding Simulation Results. Experimental results for entropy coding of the feature maps are given in Fig. 7. R_{spat} is shown as a function of n and m for no coding, and for coding of differences using an adaptive variable length entropy coding technique defined in [11]. Also shown in Fig. 7 is the unachievable performance bound for coding differences with the assumption that the source has zero memory.

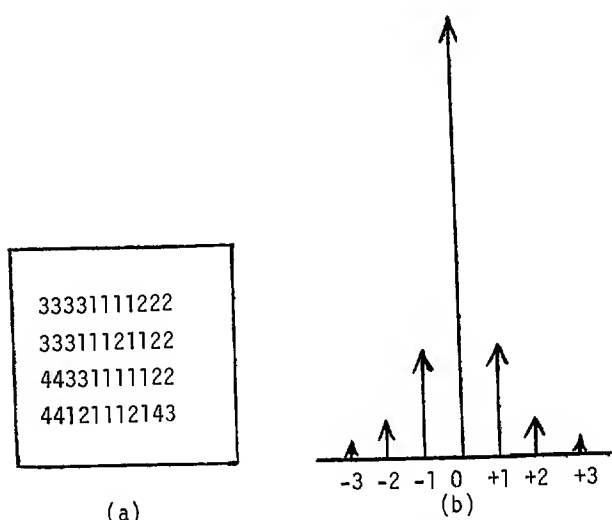


Fig. 6. (a) Partial example of typical feature map. (b) Example of typical distribution of differences between adjacent elements in feature map.

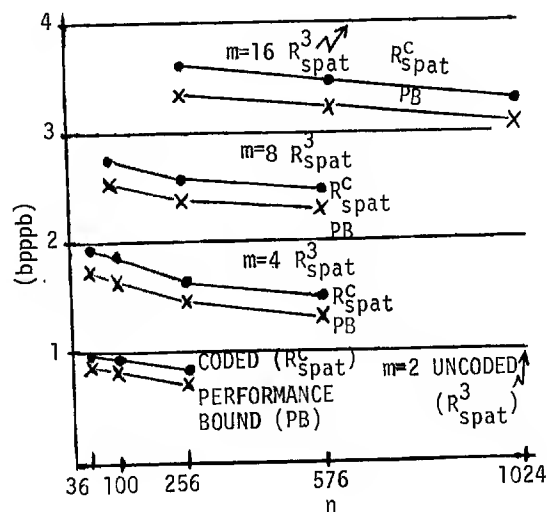


Fig. 7. Uncoded and entropy coded spatial rate vs. n for $m=2, 4, 8, 16$. Also a performance bound for coding the distribution of differences.

All curves in Fig. 7 represent the average results for all square arrays with n picture elements in the 4-band test image shown in Fig. 14(a). Figure 7 shows that entropy coding is more effective for larger n . In addition, Fig. 7 shows the entropy coding technique described in [11] and used in these simulations performed close to the unachievable performance bounds (for the zero memory source). This entropy coding is easy to implement, especially for the small source alphabet sizes of the feature maps.

Other Feature Map Coding. The above entropy coding did not introduce any degradation in representing the feature map. However, in some applications only certain spatial characteristics of the feature map are of interest. For example, only closed boundaries in the feature map may be desired in field image interpretation. The detection of spatial characteristics in $\{X_i\}_{i=1}^n$ has been simplified, since the clustering has reduced the d -dimensional image to an array of scalar numbers indicating the spatial occurrence of multidimensional features. When only spatial boundaries are of interest, the spatial encoder must preserve only those numbers in the feature map needed to identify the boundaries, and then the resulting differences in the feature map will encode to a lower spatial rate.

Another simple example of spatial redundancy removal is the deletion of subsets of the feature map with reconstruction by linear interpolation. This approach might be used if spatial resolution is higher than necessary to observe texture required in a particular application. The spatial rate in this case is simply reduced by the subsampling factor.

Coded Feature Map Rate. Let R_{spat}^C represent the rate in bpppb for the encoded feature map sequence, C_r . Define the compression ratio CR as

$$CR = \frac{R_{\text{spat}}}{R_{\text{spat}}^C} \quad (4)$$

If only natural coding of the source extension is used, the CR is nearly one. In general, the rate for the coded feature map sequence C_r is

$$R_{\text{spat}}^C = \frac{\log_2 m}{d \cdot CR} \quad (5)$$

BASIC CLUSTER COMPRESSION ALGORITHM

Total Rate

A basic configuration of the Cluster Compression Algorithm (CCA) is depicted in Fig. 8. The input to the CCA is the user supervision, which for a fixed rate mode may be only the constant number of clusters to be used for every input sequence. For a quality controlled mode, the supervision input consists of simple threshold values which control the splitting and combining of clusters in the adaptive clustering, thereby determining the number of clusters needed for each input sequence. The CCA output for each input sequence consists of the spectral and spatial definitions for which the rates are defined respectively in (2) and (5). Thus the total rate R_{tot} is given by

$$R_{\text{tot}} = R_{\text{spec}} + R_{\text{spat}}^C = \frac{b \cdot m}{n} + \frac{\log_2 m}{d \cdot CR} \quad (6)$$

Equation (6) is still valid if the entropy coder is not used, but then CR is approximately one. Note that either R_{spec} or R_{spat}^C could be zero in some applications, but in general R_{tot} is a mixture of both spectral and spatial definitions. Let R_{tot} in (6) be equal to a constant R_{tot}^* , and express n as a function of m to get

$$n = b \cdot m / [R_{\text{tot}}^* - \log_2 m / (d \cdot CR)] \quad (7)$$

where

$$R_{\text{tot}}^* > \frac{\log_2 m}{d \cdot CR} \quad (8)$$

A plot of n vs. m in (7) for a constant $R_{\text{tot}}^* = 1$ is given in Fig. 9 with $b=6$, $d=4$ and $CR=1$ and 2. This plot demonstrates that a given R_{tot} may be approximately obtained by many combinations of m , n and CR , and similarly there is much flexibility in choosing the division of R_{tot} between the spectral and spatial definitions.

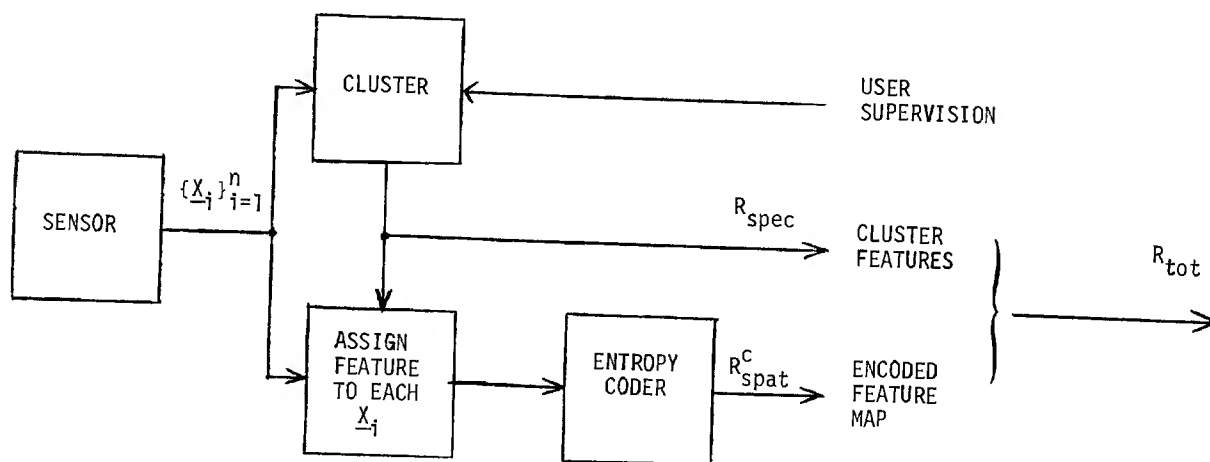


Fig. 8. The Basic Cluster Compression Algorithm

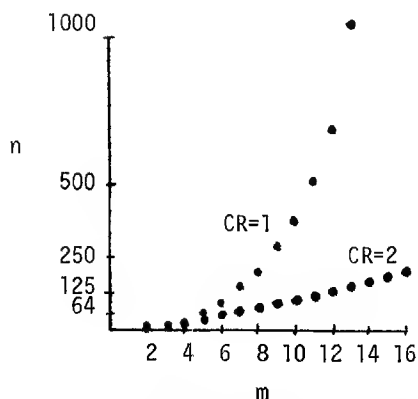


Fig. 9. Plot of n vs. m for $b=6$, $d=4$ and $R_{tot}=1$.

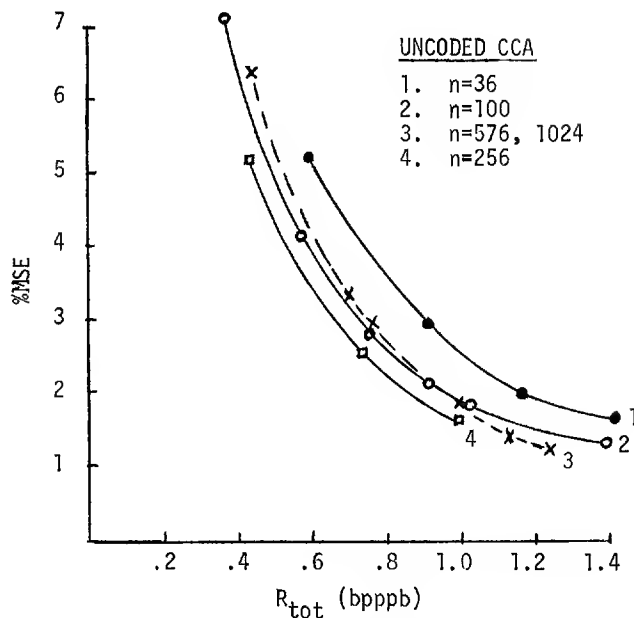


Fig. 10. %MSE vs. R_{tot} for various values of n for the uncoded CCA

Performance vs. Rate

In terms of implementation it is desirable to keep both m and n small, since clustering requires more computations and memory when m and n are larger. However, one must also question the comparative impact on performance of various (m, n, CR) combinations which have the same R_{tot} . Performance might be measured subjectively in terms of image appearance, or quantitatively in terms of percent mean square error (%MSE), or classification accuracy. Some pictures are provided at the end of this paper for subjective evaluations of image quality, but at this point a quantitative measure of quality is desired. Classification accuracy is an important quantitative measure of image quality for many ERTS applications. However, classification accuracy is dependent on the classification technique, as well as the compression technique, and often significant accuracy improvements can be obtained by tailoring the classification technique for each specific compression technique. Results of jointly investigating corresponding compression and classification techniques will appear in a later publication. Percent MSE is often not a good performance criteria when comparing across different compression techniques, or different images, but %MSE is generally meaningful for comparing the performance from options on the same compression technique and same test image. Thus %MSE is used here as a performance criteria in comparing (m, n, CR) combinations.

Let \underline{X} represent the original d -dimensional image data, and let $\hat{\underline{X}}$ represent the reconstructed approximation of the same image data after compression. Then the MSE between the original image and the approximate image is the expected value of the square of the Euclidean distance between \underline{X} and $\hat{\underline{X}}$.

$$MSE = E[||\underline{X} - \hat{\underline{X}}||^2] \quad (9)$$

The average energy, \mathcal{E} in the original image is defined in terms of the variance in the i th spectral band, σ_i^2 , $i=1,2,\dots,d$.

$$\mathcal{E} = \sum_{i=1}^d \sigma_i^2 \quad (10)$$

Thus the %MSE is defined by

$$\%MSE = \frac{MSE \cdot 100}{\mathcal{E}} \quad (11)$$

Simulation ResultsUncoded CCA

Various configurations of the CCA were simulated on the test image shown in Fig. 14(a). This test image is a 256 x 256 picture element subset from a 4-band ERTS image of the Verde Valley. The purpose of these simulations is to obtain a measure of performance vs. rate as a function of the (n, m, CR) combinations. The first simulations used the Basic Clustering Algorithm (BCA) of Fig. 3, but without coding of the feature map. This corresponds to determining performance vs. rate as a function of (n, m) pairs for $CR=1$. The performance of this uncoded CCA in terms of %MSE vs. R_{tot} for various values of n is plotted in Fig. 10. For each n shown, all square subsets of n picture elements in the test image were compressed using various values of m . For example, for $n=36$ the test image was processed four times as follows: two features ($m=2$) per each subset of 6×6 ($n=36$) picture elements, three features ($m=3$) per 6×6 , four features ($m=4$) per 6×6 , and five features ($m=5$) per 6×6 . These four simulation results were then plotted in Fig. 10 and a smooth curve was drawn between them. The cluster feature used for this image approximation application was only the cluster mean, and each cluster mean was defined by either 6 or 8 bits per band ($b=6$ or 8). Therefore, the image approximation consisted of substituting the closest cluster mean for each picture element. For every simulation with a given n and m , the %MSE between the original and reconstructed image was calculated from (9)-(11), and R_{tot} was calculated from (6). For $n=576$ and $n=1024$ the performance curves were essentially identical. The results in Fig. 10 show that for the selected source data the uncoded CCA peaks in performance for all R_{tot} at n approximately equal to 256. Similar results have also been obtained from simulations with other test images.

Coded CCA

The same simulations that were conducted for the uncoded CCA ($CR=1$) were also conducted with entropy coding of the feature map. This simulation configuration is called the coded CCA. The purpose of these simulations was to measure performance vs. rate as a function of (m, n) pairs when CR is determined by the entropy coding of the feature map differences. The entropy coding technique used is that which was discussed earlier and which is defined in [11]. Basically the encoder adaptively assigns variable length codewords to the differences between adjacent elements of the feature map, which results in reducing the average rate while allowing the feature map to be exactly reconstructed. All overhead rate costs have been included in the compression ratio results for the entropy coding. The spatial rate reduction, or CR due to entropy coding was shown in Fig. 7 to be greater for larger n . Thus the performance curves for the coded CCA will shift more to the left of the uncoded CCA curves when n is larger. This effect is observed by comparing the coded CCA curves in Fig. 11 with the uncoded CCA curves in Fig. 10. The coded CCA continues to increase in performance as n increases, but it is near peak performance for $n=256$.

Adaptive CCA

In the next simulations, the Adaptive Clustering Algorithm (ACA), which was discussed under Cluster Approaches, was used to adaptively determine the number of clusters to use for each sequence of n picture elements. The total rate for this CCA configuration is approximately the average of all total rates for each of the sequences of n elements, since the overhead to identify the number of clusters for each sequence is very small. The advantage of the adaptive CCA is that a lower average value of m can typically be used to obtain the same image quality. The performance and rate of the adaptive CCA are now dependent on the criteria for increasing or decreasing the number of clusters, as well as the combination of (n, m, CR) . The performance of the adaptive CCA was observed to be less dependent on the size of n , with all results for n of ten or greater nearly on the same performance curve. Although R_{tot} was sensitive to

changes in the criteria for determining the number of clusters used per sequence, the performance vs. rate trade-off was very insensitive to changes in that criteria. There is a significant performance gain in the adaptive CCA relative to the coded and uncoded CCA. This gain is shown in curves 3, 4 and 5 of Fig. 12. However, the performance gain due to adaptability is dependent on both the image data and user application.

Hadamard and Fourier Transform Techniques

Tests were performed to compare the MSE performance of the CCA with some other compression techniques which are well known. The two techniques used for comparison are the well known Hadamard and Fourier transform techniques. A discussion of Hadamard and Fourier transforms in image coding is given in [12] and [13]. Both techniques were used in an adaptive form, and the compression was done on each band of the test image independently. Some improvement in performance could be obtained by modifying the technique to work jointly on all four bands. The Hadamard transform technique involved taking the two dimensional Hadamard transform of all 8×8 subsets of the image. Then the 64 coefficients were placed in 9 zones, each of which had the quantization and corresponding bit rate per coefficient based on the variance, or energy of the coefficients in the zone. The Fourier transform technique was similarly adaptive, but in addition a symmetrical transform approach was used which doubly folded each data subset to provide horizontal and vertical symmetry. This symmetry reduces the intensity discontinuities at the boundaries due to the low pass filtering. The %MSE vs. total rate for these two adaptive transform techniques is shown in curves 1 and 2 of Fig. 12 along with the CCA performance curves.

Simulation Summary

In Fig. 13 a summary of the above simulation results is given by plotting R_{tot} required for 2 %MSE vs. n for the various compression configurations. The data points plotted in Fig. 13 were interpreted from the smooth curves in the previous figures, as well as from other simulation results. These curves point out a substantial MSE penalty for $n < 100$ and a good value for n would be 256. Also simulated was the use of the absolute value distance measure in the clustering algorithm. It was observed that both %MSE and subjective image appearance were essentially unchanged relative to the use of the Euclidean distance measure. This result could be significant from an algorithm implementation standpoint.

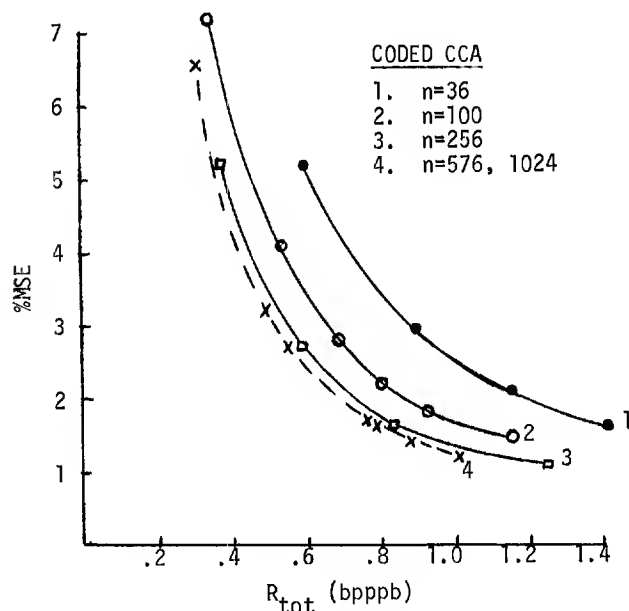


Fig. 11. %MSE vs. R_{tot} for various values of n for the coded CCA

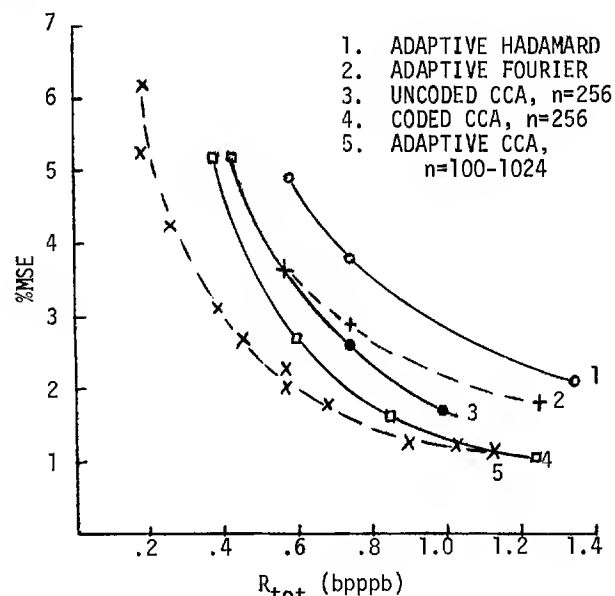


Fig. 12. %MSE vs. R_{tot} for various CCA configurations and for adaptive Hadamard and Fourier techniques.

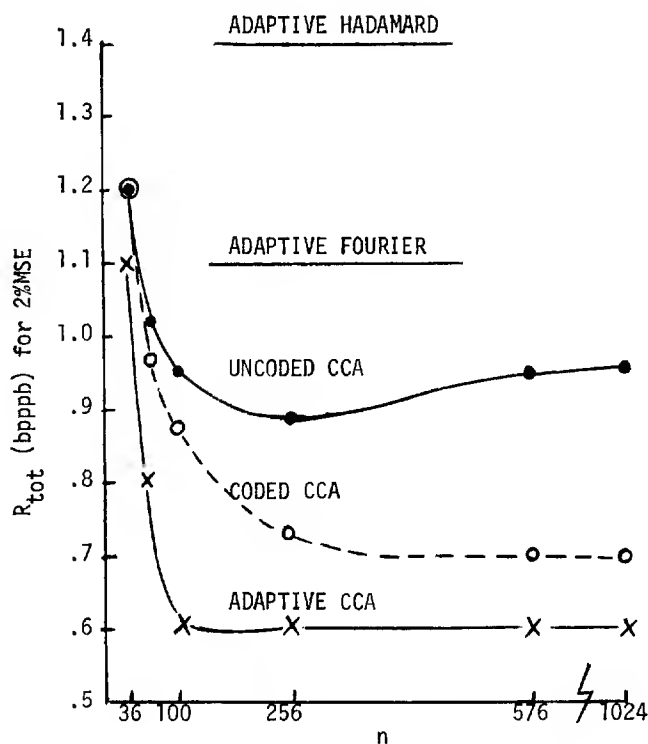


Fig. 13. R_{tot} for 2%MSE vs. n for various CCA configurations and for adaptive Hadamard and Fourier techniques

The %MSE vs. R_{tot} performance results agreed well with the subjective measure of image appearance for the three CCA forms simulated above. This would be expected since the uncoded, coded and adaptive CCA all introduce the same type of image degradation. In particular the coding used in the coded CCA changes the spatial data rate, but does not change the image quality. However, it should not be assumed that the results in Fig. 13 apply to all CCA forms. There exist many forms of the CCA with different types of image degradation due to using different types of feature map encoding. For example, one could use subsampling of the feature map to obtain CR=2, 4, or 8 and this would greatly change the type of image degradation, as well as the performance as a function of (n,m) pairs. In fact simulations at low R_{tot} have shown that a CCA with feature map subsampling gives an image of much better appearance than that of an adaptive CCA, even though both CCA results were at the same %MSE and R_{tot} . In general, there exist many options for feature map encoding, and the corresponding CCA forms often have trade-offs between spectral and spatial quality which may not be meaningfully measured in terms of MSE. However, the results summarized in Fig. 13 do define meaningful relative performance variations due to (m,n) pairs for useful CCA forms of no coding, entropy coding, and adaptive clustering.

Other CCA Configurations

There are many possible variations in the form of the cluster compression concept. The above simulations demonstrated variations in the clustering and in the feature map coding. Other variations could similarly center about coding sequences of the features extracted from successive local sources of the image, in order to reduce R_{spec} . When clustering sequences of local sources, there is likely to be considerable correlation in cluster descriptions from the different sources. Another method of removing this redundancy is to collect the cluster descriptions for a sequence of local sources and then cluster or

combine the local source clusters into a smaller set of clusters. The number of clusters per local source might not be changed, but the number per sequence could be considerably reduced. The clusters in each local source are a subset of the sequence clusters. Cascaded clustering also allows for modification of local clustering based on knowledge of clustering results for surrounding local sources. This variation of the CCA is called the cascaded CCA and it has been previously discussed in [1]. The cascaded CCA referred to in the following image reconstruction simulations, simply uses the BCA to cluster each local source, and the BCA again to cluster the cluster means of each cluster in the sequence of local sources. Better results could be obtained from using intercluster distance measures and other cluster features in cascaded clustering.

Reconstructed Image Examples

Figures 14-19 provide examples of reconstructed images resulting from simulations of the various CCA forms. The image reconstruction consisted simply of substituting the appropriate cluster mean for each picture element. In general, the CCA provides the compressed data in a form which enables simple decompression and selective access from the data in applications of image approximation, or computer interpretation.

Conclusion

The Cluster Compression Algorithm was shown to be an attractive new approach to multispectral image data compression. The coded and adaptive CCA were shown to provide substantial performance gains over the uncoded CCA. The entropy coding increased in effectiveness for larger n , but overall performance in terms of %MSE vs. R_{tot} approached near maximum performance for $n \geq 16$ with non-adaptive clustering, and for $n \geq 10$ with adaptive clustering. Other variations of the CCA, such as feature map subsampling and the cascaded CCA were also discussed. Picture results from simulations of all the CCA variations were provided for subjective image quality evaluation. These pictures and others demonstrated the CCA to be capable of excellent performance under both MSE and subjective fidelity criteria.

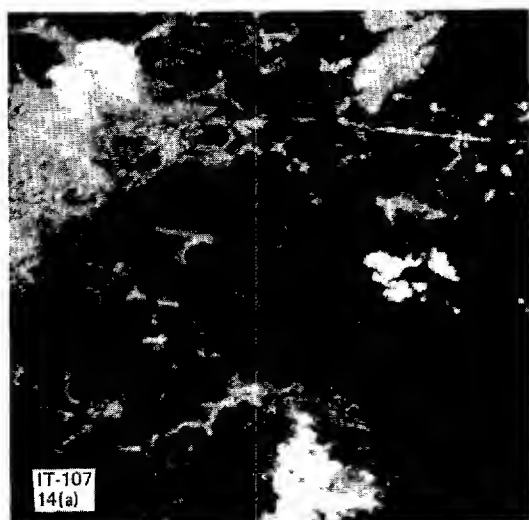
Acknowledgement

The author is indebted to J. E. Getz and S. S. Salisbury for providing the computer simulation support and to R. F. Rice, Dr. C. Wu and R. Piereson for their suggestions.

References

1. E. E. Hilbert, "Joint Classification and Data Compression of Multidimensional Information Sources-Application to ERTS," Internat. Conf. on Commun., Vol. II, June 1975.
2. L. A. Zadeh, "Fuzzy Sets," Inform. Control, Vol. 8, pp. 338-353, 1965.
3. J. N. Gupta and P. A. Wintz, "A Boundary Finding Algorithm and its Applications," IEEE Trans. on Circuits and Systems, Vol. CAS-22, No. 4, April 1975.
4. E. P. F. Kan, "Data Clustering: An Overview," Lockheed Electronics, Co., Inc., HASD, Houston, Texas, Tech. Rep. 640-TR-080, March 1972.
5. L. Kanal, "Patterns in Pattern Recognition: 1968-1974," IEEE Trans. Inform. Theory, Vol. IT-20, pp. 697-722, Nov. 1974.

6. D. J. Hall et al, "Development of New Pattern-Recognition Methods," Aerospace Research Laboratories, ARL 73-0153, Nov. 1973.
7. G. H. Ball and D. J. Hall, "ISODATA, A Novel Method of Data Analysis and Pattern Classification," Stanford Res. Inst., Menlo Park, Calif., Tech. Rep. AD699616, Apr. 1965.
8. E. P. F. Kan, "The Latest Version of ISODATA (A)/ISOCLS," Lockheed Electronics Co., Inc., HASD, Houston, Texas, Tech. Memo TM642-570, Sept. 1972.
9. E. P. F. Kan, "The JSC Clustering Program ISOCLS and Its Applications," Lockheed Electronics Co., Inc., HASD, Houston, Texas, Tech. Rep. 640-TR- , July 1973.
10. P. H. Swain, "Pattern Recognition: A Basis for Remote Sensing Data Analysis," Lab. for Applic. of Remote Sensing, Purdue Univ., W. Lafayette, Ind., LARS Inform. Note 111572, Sept. 1973.
11. R. F. Rice and J. R. Plaunt, "Adaptive Variable-Length Coding for Efficient Compression of Spacecraft Television Data," IEEE Trans. Commun., Vol. COM-19, No. 6, Dec. 1971.
12. W. K. Pratt, J. Kane, and H. C. Andrews, "Hadamard Transform Image Coding," Proc. IEEE, Vol. 57, No. 1, Jan. 1969.
13. A. Habibi and P. A. Wintz, "Image Coding by Linear Transformation and Block Quantization," IEEE Trans. Commun. Technol., Vol. COM-19, pp. 50-62, Feb. 1971.



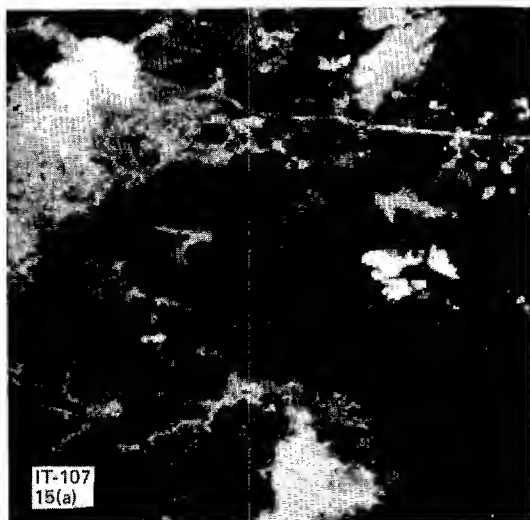
(a) 6 bpppb



(b) 8 bpppb

Fig. 14. Original pictures of 256 x 256 picture elements.

- (a) A subset of an ERTS image of the Verde Valley; band 2 selected for display from a total of 4 bands, quantized to 6 bits per picture element per band (bpppb).
- (b) Red band of girl selected from a total of 3 bands, quantized to 8 bpppb.



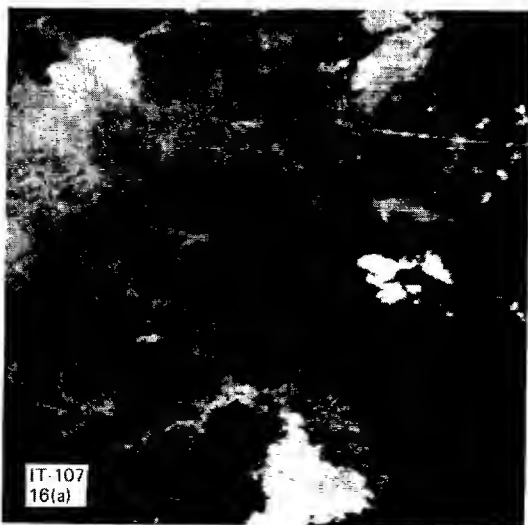
(a) .88 (uncoded), .8 (coded) bpppb



(b) 1.18 (uncoded), 1.07 (coded) bpppb

Fig. 15. Examples of the uncoded and coded cluster compression algorithm (CCA). The local source subsets are 10 x 10 picture elements ($n=100$) and each is clustered into 5 clusters ($m=5$). All overhead is always included in the rate computations.

- (a) Spectral rate (R_{spec}) = .3, uncoded spatial rate (R_{spat}^3) = .58, coded spatial rate (R_{spat}^c) = .5, total rate (R_{tot}) = .88 (uncoded), .8 (coded) bpppb.
- (b) R_{spec} = .4, R_{spat}^3 = .776, R_{spat}^c = .666, R_{tot} = 1.18 (uncoded), 1.07 (coded) bpppb.



(a) .79 (uncoded), .68 (coded) bpppb



(b) 1.16 (uncoded), 1.12 (coded) bpppb.

Fig. 16. Examples for the adaptive CCA. The local source subsets are 10×10 picture elements ($n=100$) and the number of clusters (m) for each is adaptively determined.

- (a) $m=1$ to 15 with an average of 4.5; $R_{\text{spec}} = .28$, $R_{\text{spat}}^3 = .51$, $R_{\text{spat}} = .4$, $R_{\text{tot}} = .79$ (uncoded), .68 (coded) bpppb.
- (b) $m=1$ to 16 with an average of 5.6; $R_{\text{spec}} = .46$, $R_{\text{spat}}^3 = .7$, $R_{\text{spat}} = .66$, $R_{\text{tot}} = 1.16$ (uncoded), 1.12 (coded) bpppb.



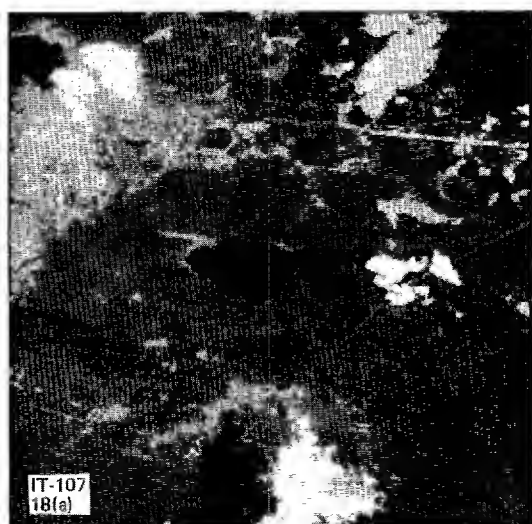
(a) .97 (uncoded), .72 (coded) bpppb



(b) .58 (SS2) bpppb

Fig. 17. Another example of the adaptive CCA which also demonstrates combined use of subset sampling by $1/2$ (SS2) of the feature map. The local source subsets are 16×16 picture elements ($n=256$), and the number of clusters (m) equals 1 to 19 with an average of 5.57.

- (a) $R_{\text{spec}} = .18$, $R_{\text{spat}}^3 = .79$, $R_{\text{spat}} = .54$, $R_{\text{tot}} = .97$ (uncoded), .72 (coded) bpppb.
- (b) $R_{\text{spec}} = .18$, $R_{\text{spat}}^3 = .4$ (SS2), $R_{\text{tot}} = .58$ (SS2) bpppb.



(a) .664 (uncoded), .519 (coded) bpppb



(b) .357 (SS2) bpppb

Fig. 18. Example of cascaded CCA incorporating no coding, coding, and subset sampling by 1/2 of the feature map. The local source subsets are 16 x 16 picture elements ($n=256$) which are initially clustered into 8 clusters each. The cascaded clustering combines 128 clusters from a sequence of 16 local source subsets into 16 clusters per sequence and an average of 5.44 clusters per subset ($m=5.44$).

(a) $R_{\text{spec}} = .047$, $R_{\text{spat}}^3 = .617$, $R_{\text{spat}}^C = .472$,
 $R_{\text{tot}} = .664$ (uncoded), .519 (coded) bpppb.

(b) $R_{\text{spec}} = .047$, $R_{\text{spat}}^3 = .31$ (SS2),
 $R_{\text{tot}} = .357$ (SS2) bpppb.



(a) .583 (uncoded), .403 (coded) bpppb



(b) .183 (SS4) bpppb

Fig. 19. Example of cascaded CCA incorporating no coding, coding, and subset sampling by 1/4 (SS4) of the feature map. The local source subsets are 16 x 16 picture elements ($n=256$) which are initially clustered into 4 clusters each. The cascaded clustering combines 64 clusters from a sequence of 16 local source subsets into 16 clusters per sequence and an average of 3.16 clusters per subset ($m=3.16$).

(a) $R_{\text{spec}} = .05$, $R_{\text{spat}}^3 = .533$, $R_{\text{spat}}^C = .353$,
 $R_{\text{tot}} = .583$ (uncoded), .403 (coded) bpppb.

(b) $R_{\text{spec}} = .05$, $R_{\text{spat}}^3 = .133$ (SS4),
 $R_{\text{tot}} = .183$ (SS4) bpppb.

A DUAL MODE NONLINEAR INTERPOLATIVE COMPRESSOR FOR SAR IMAGES

Dr. Wen-hsiung Chen
 Senior Engineering Specialist
 Aeronutronic Ford Corporation
 Western Development Laboratories Division
 3939 Fabian Way
 Palo Alto, California 94303

Abstract

A dual mode nonlinear interpolative compressor is described which compresses synthetic aperture radar images. The compression algorithm takes advantage of (1) the strong contrast between the target and the background, (2) the small percentage of target area compared with the total SAR image. The algorithm operates in two modes, i.e., an exact data transmission mode and an interpolative mode. The image is first segmented into small sub-blocks. The compressor then selects one of two modes according to the pixel values inside the block. If any pixel inside the block is above a predetermined threshold all pixels in this block are transmitted. On the other hand, if all pixels inside the block are below the threshold only one pixel is transmitted. At the receiver a two dimensional interpolation is then conducted to estimate the entire pixels in the block. Under the stationary assumption of the background information a second order statistic of the SAR image is used to derive the interpolative scheme. The compressor not only offers a high degree of exact target recovery but also offers a good recovery of the background information. Computer simulation results indicate that a good quality reconstruction can be accomplished with 1.5 bits per pixel.

Introduction

In SAR images the signal intensity in the target area is usually very much higher than that of the background. Also, the target usually occupies a relatively small area when compared with the background. Assuming that the main objective to compress SAR images is to recover the target area as close to the original as possible. Then the transform coding technique based on the average statistics offers little advantage since the average statistics collected from SAR images are contributed most by the background. Unless a separate processing of the target and the background incorporate with a complicated adaptive coding scheme is used, the transform coding tends to recover the background fairly well but distorts the target rather severely. With this observation, a dual mode nonlinear interpolative compressor is introduced. The compressor not only offers a high degree of exact target recovery incorporate with a good background recovery, but also offers a simple way of hardware implementation.

The dual mode nonlinear interpolative compressor operates the spatial data in two distinct modes. For those data above a predetermined threshold all data are transmitted. For those data below the threshold only a few samples are transmitted and at the receiver a nonlinear interpolation is performed to reconstruct the data. Suppose that the target area occupies less than 15% of a SAR image, then a data compression ratio of at least 5 to 1 can be achieved.

It is well understood that redundancy removal techniques such as that to be used here, tend to propagate errors. However, if an error correcting code is used in the mode specification, a reliable data transmission is feasible. The mode error corrections will cause little overhead to the compressor.

Statistical Properties of SAR Images

Statistical information plays an important role in developing an efficient algorithm. For a better understanding, some of the statistics which are of importance to the dual mode nonlinear interpolative compressor are briefly presented.

Figure 1 shows a typical probability density function of SAR images. (The upper end of the histogram which represents target information has been magnified somewhat.) The histogram has two peaks where the strong low-grey-level peak corresponds to clutter, background, etc., and the weak high-grey-level peak corresponds to bright target areas. Generally, the separation between the two peaks is very high and the overlap is low. Since the low-grey-level background dominates most of the image, the average intensity is expected to be very low. For a typical SAR image with grey levels of 0 to 255 (8 bits data) the mean is found to be 5.68. Figure 2 shows a plot of the percentage of pixels lying below a magnitude threshold level for a typical SAR image. A strong low level distribution of the SAR image is clearly demonstrated.

A series of experiments has been conducted to find the second order statistics of SAR images. With a stationary assumption of SAR images the autocorrelation function is found to be closely related to a double-sided exponential function which can be modeled by

$$R(\tau) = e^{-\alpha |\tau|} \quad (1)$$

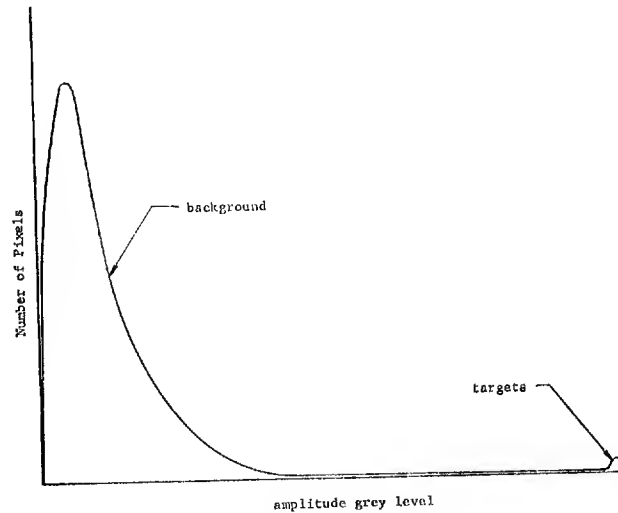


Figure 1 Histogram Distribution of a Typical SAR Image

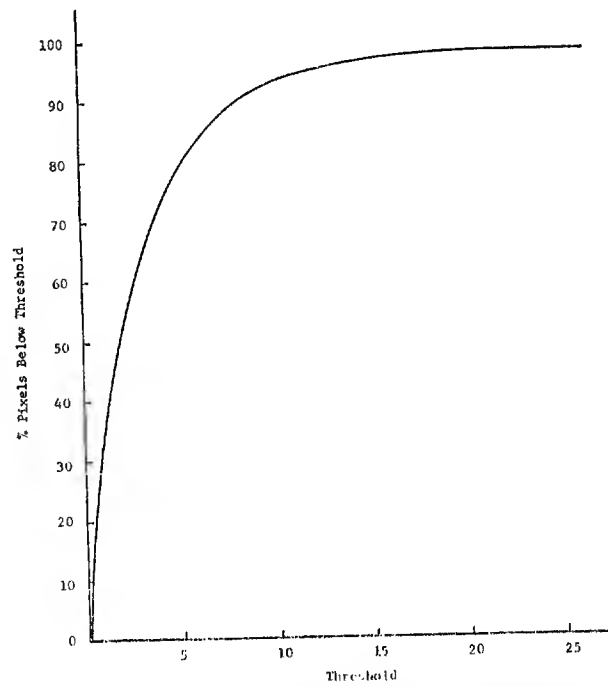


Figure 2 Number of Pixels Below Threshold Versus Threshold Level for a Typical SAR Image

Figure 4 illustrates an experimental autocorrelation function for a typical SAR image. The modeling of this function by equation (1) is also plotted. The close fitting of the experimental data to equation (1) suggests that the image pixels along rows or columns are highly correlated and can be modeled by the first order Markov process.

Compression Algorithm

The compression algorithm used here is based on the estimation of most background information through a two dimensional nonlinear interpolation. The image is first segmented into small square blocks of pixels. The compressor then selects one of two modes according to the pixel values inside the block. If any pixel inside the block is above a predetermined threshold all pixels in this block are transmitted. On the other hand, if all pixels inside the block are below the threshold, only one pixel in this block is transmitted. At the receiver, a nonlinear interpolation is then conducted to estimate the entire pixels in the block. Figure 3 illustrates a segmentation of image pixels into $M \times M$ blocks. Suppose that every pixel in the first block is below a threshold, then only $f(1,1)$ is transmitted. At the receiver $f(1,2)$, $f(1,3) \dots f(1,M)$ are estimated from $f(1,1)$ and $f(1,M+1)$. Next $f(M+1,2)$, $f(M+1,3) \dots f(M+1,M)$ are estimated in a similar way using $f(M+1,1)$ and $f(M+1,M+1)$. (The estimation is not necessary if all pixels in this block are transmitted.) The pixel values in these two lines are then used to estimate all the values along column directions to fill out the entire block.

The interpolation scheme used is to optimize the mean square error between the original data and the estimated data. Suppose that an estimation of $f(t)$, $0 < t < T$, in terms of $f(0)$ and $f(T)$ is required, i.e.,

$$\hat{f}(t) = a f(0) + b f(T) \quad (2)$$

Then the error must be orthogonal to $f(0)$ and $f(T)$, i.e.,

$$E \{ [f(t) - \hat{f}(t)] f(0) \} = 0 \quad (3)$$

and

$$E \{ [f(t) - \hat{f}(t)] f(T) \} = 0 \quad (4)$$

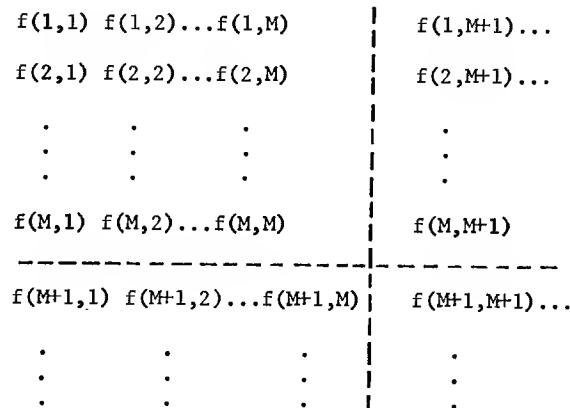


Figure 3 Image Segmentation of $M \times M$ Blocks

Substitution of equation (2) into (3) and (4) and solving these equations one obtains

$$a = \frac{R(0)R(t) - R(T)R(T-t)}{R^2(0) - R^2(T)}$$

$$b = \frac{R(0)R(T-t) - R(t)R(T)}{R^2(0) - R^2(T)}$$

where R is the autocorrelation function. With the autocorrelation function modeled by $R(T) = e^{-\alpha |T|}$ there results

$$a = \frac{\sinh \alpha (T-t)}{\sinh 2T}$$

$$b = \frac{\sinh \alpha t}{\sinh \alpha T}$$

The bit compression factor can be computed from the relative time that the compressor is in the interpolative mode. Let the original image be quantized in B bits and let P be the probability of being in the interpolative mode. Then the average code bits required in a block of $M \times M$ pixels is

$$\text{Average code bits} = P(B + 1) + (1 - P)(M^2 B + 1) \quad (5)$$

In the above equation, an extra one bit is assigned to the mode specification. With total code bits of $M^2 B$ per block the bit compression factor is found to be

$$\text{BCF} = \frac{M^2 B}{P(B + 1) + (1 - P)(M^2 B + 1)} \quad (6)$$

For typical SAR images since the target area generally occupies only a small portion of the image the probability P is expected to be very high. Let P be assumed to be 0.85. Then with an 8 bits input data and a 5 by 5 data block segmentation the bit compression factor is 5.3. For an extreme case, a maximum bit compression factor of 22.2 can be achieved. This is equivalent to the case where there are no metal objects in the radar return. The optimum block size for the compressor is dependent upon the image resolution and the target distribution inside the image. No effort is intended here to find this value.

The limitation of the dual mode compression system is that the system tends to propagate channel error. One way to minimize this problem is to apply an error correcting code to those bits which specify the mode transitions. The increase in bit assignment for mode representation will not have too much effect on the overall bit compression factor. As an example, if 5 bits are used in the mode specification, then the bit compression factor becomes

$$\text{BCF} = \frac{M^2 B}{p(B + 5) + (1 - p)(M^2 B + 5)} \quad (7)$$

With p , B , and M equal to 0.85, 8 and 5, respectively, the bit compression factor is 4.8 which is only slightly less than the case when only one bit is used as mode specification.

Experimental Results

A computer simulation has been performed to evaluate the performance of the dual mode nonlinear interpolative compressor to the SAR images. Figures 7(a) and 7(b) contain two typical 8 bits SAR images of 256 by 256 pixels as an input to the dual mode compressor. With the help of figures 1 and 2 a number of threshold levels has been determined and a number of data compression performed. The block size of the compressor is arbitrarily chosen to be 5 by 5. Figures 8(a) to 8(h) illustrate some of the experimental results. A good reconstruction of 1.5 bit pictures are demonstrated with virtually no target degradation. Figure 5 shows the mean square error performance of the image in figures 7(a) and 7(b) as a function of average code bit. The mean square error plotted can be considered very good since they are normalized by the dim signal energy. Also plotted in figure 5 is the mean versus average code bit. As expected, the interpolation tends to drop the average signal intensity in a minor degree. Figure 6 shows a plot of the

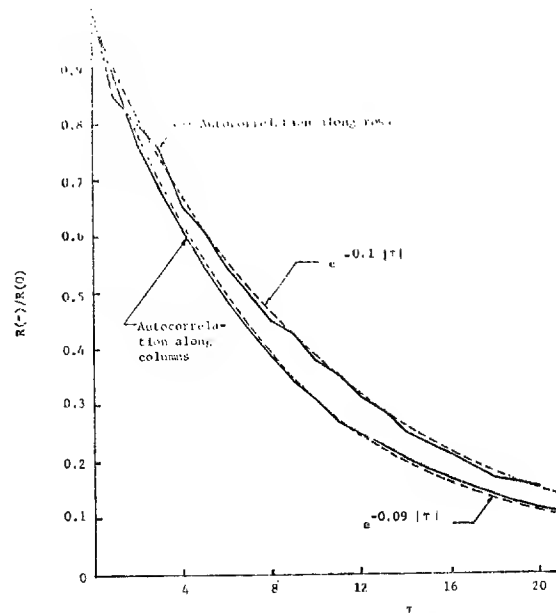


Figure 4 Normalized Autocorrelation Function of a Typical SAR Image

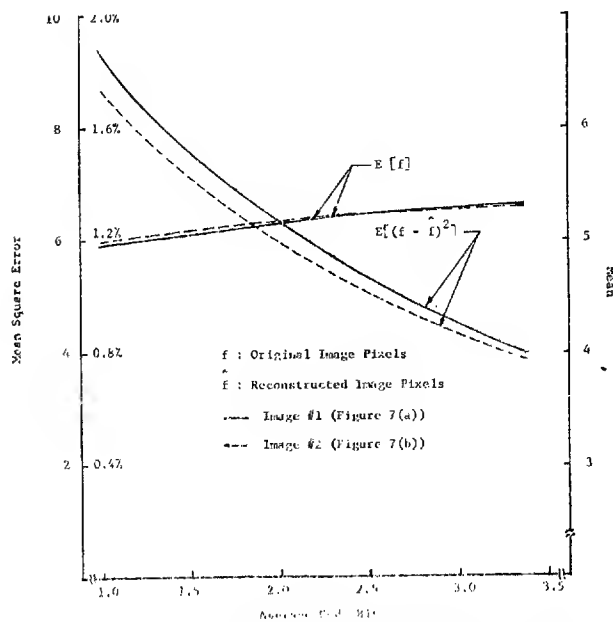


Figure 5 Mean Square Error/Mean as a Function of Average Code Bit

average code bit as a function of the threshold level for the image in figure 7(a). Once a threshold level is determined the bit compression factor can readily be calculated.

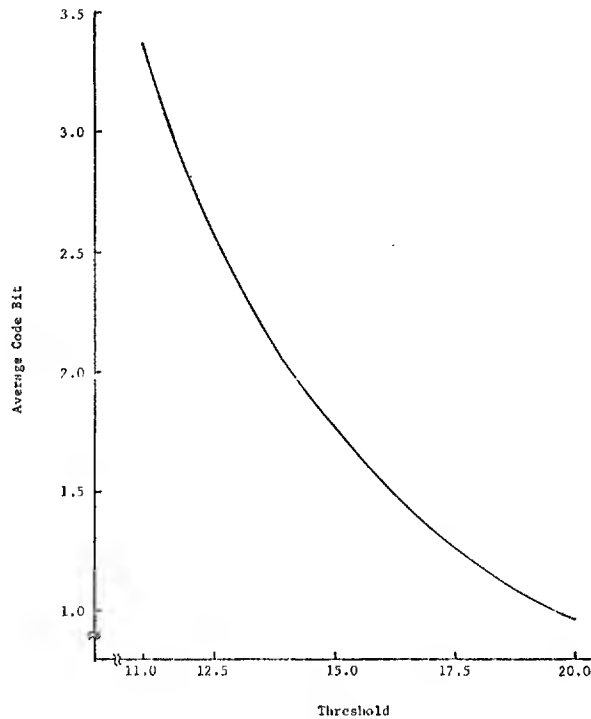


Figure 6 Average Code Bit Versus Threshold Level

Summary

The dual mode compressor has proven to provide a substantial data compression for the SAR image. The compression algorithm has an exact data transmission mode and an interpolative mode. The compression is essentially done in the interpolative mode. Due to the specific statistical characteristics of the image a nonlinear interpolation of the data points can be derived. The technique can be considered excellent for SAR image applications from the standpoint of implementation and performance.

References

1. C. M. Kortman, "Redundancy Reduction - A Practical Method of Data Compression," Proceedings of the IEEE, Vol. 55, No. 3, March 1967, pp. 253-263.
2. D. Hochman, H. Katzman, and D. R. Weber, "Application of Redundancy Reduction to TV Bandwidth Compression," Proceedings of the IEEE, Vol. 55, No. 3, March 1967, pp. 263-266.
3. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill Inc., 1965.

A FAST TWO-DIMENSIONAL KARHUNEN-LOEVE TRANSFORM

Robert M. Haralick
 Norman Griswold
 Nimitra Kattiyakulwanich
 University of Kansas Center for Research, Inc.
 Remote Sensing Laboratory
 Lawrence, Kansas 66045

Abstract

One frequently used image compression method is based on transform coding. In terms of RMS error, the best transform is the Karhunen-Loeve (Principal Components). This method is not generally used due to computational complexity. In this paper we show that under isotropic conditions the Karhunen-Loeve is almost separable and that an approximate fast principal components transform exists. Our results indicate that the fast K-L is nearly as good as the true K-L and that it yields better results than other discrete transforms such as DLB, SLANT, or Hadamard. The approximations and errors are discussed in terms of the RMS and RMS correlated error.

Introduction

The purpose of transform coding is to store or represent data in a reduced dimensional space and yet preserve the data structure. When mean square error is the optimality criterion, the principal components or Karhunen-Loeve expansion is the best. However, the principal components technique is generally not used in transform coding image compression work because of its computational complexity. In this paper we describe how to implement the principal components or Karhunen-Loeve transform (KLT) as a fast transform for image data when the image data satisfies mild stationarity and isotropic conditions.

Figure 1 illustrates the typical way transform coding is done. The N_r row and N_c columns image is partitioned into subimages or windows each having K_r rows and K_c columns. There are $(N_r/K_r) \cdot (N_c/K_c)$ such subimages. Each subimage is transformed using a Hadamard, Fourier, Slant, Discrete Cosine, or Discrete Linear Basic transform (all of which have fast implementations) or by the principal components or Karhunen-Loeve transform (which is slow). Those transform domain components which have the highest energy or variance are quantized and stored or transmitted while those transform domain components which have lowest energy or variance are not retained and effectively set to zero. Data compression is achieved because the number of bits required to encode the highest energy transform components are much less than the number of bits required to encode the data in its original spatial form.

In order to achieve the data compression by the principal components technique, the grey levels in each of the $(N_r/K_r) \cdot (N_c/K_c)$ subimages must be arranged as a vector and the $(K_r K_c) \times (K_r K_c)$ auto-covariance matrix for a sample fraction f these vectors must be computed. This requires

$$f \left(\frac{N_r}{K_r} \right) \left(\frac{N_c}{K_c} \right) (K_r K_c) (K_r K_c) = f N_r N_c K_r K_c$$

operations. Next the eigenvectors and eigenvalues of the auto-covariance matrix must be found. This requires on the order of $(K_r K_c)^3$ operations. To take the dot product of each subimage with $K_r K_c$ vectors to obtain the transformed image requires $(N_r/K_r) (N_c/K_c) (K_r K_c)^2$ operations. The fast transform technique we describe requires

$$(N_r/K_r) (N_c/K_c) (K_r K_c) (K_r + K_c)$$

operations. This represents a savings factor of $K_r K_c / (K_r + K_c)$

Stationarity and Isotropy

In order to begin our discussion of how a fast KLT comes about, we first must discuss what we mean by stationarity and isotropy for an image. Suppose an image I has N_r rows and N_c columns. Partition this image into mutually exclusive subimages each K_r rows by K_c columns. (We assume N_r is an integer multiple of K_r and N_c is an integer multiple of K_c). We say the image I is stationary (in the weak sense) when two conditions are satisfied:

- (1) The mean grey tone of all resolution cells situated in subimage row column coordinates (i, j) is the same constant μ which is independent of the relative subimage coordinates (i, j) .
- (2) The grey tone covariance of all pairs of resolution cells situated in subimage row column coordinates $[(i, j), (i+n_1, j+n_2)]$ is a function $\sigma(n_1, n_2)$ only of the row column translations (n_1, n_2) and independent of the relative subimage coordinates (i, j) .

As illustrated in Figure 2, condition (1) implies, for example, that the average grey tone of all resolution cells occupying the upper left hand corner of the subimages equals the average of all resolution cells occupying the upper right hand corner of the subimages. In other words, fix a relative coordinate of the subimage. Then the average grey tone taken over all resolution cells having those relative coordinates in the subimages is equal to the average taken over any other relative coordinates.

As illustrated in Figure 3, condition (2) implies, for example, that the average second moment grey tone taken between resolution cells occupying the first and second columns of the first row in each subimage must equal the average second moment grey tone taken between resolution cells occupying the fifth and sixth columns of the third row in each subimage. In other words, fix some relative coordinates of the subimage. Then choose a row column translation. Then the second moment grey tone taken between all resolution cells situated in the specified relative coordinates and in the specified relative coordinates shifted by the row and column translation is independent of the specified relative coordinates and only a function of the translation factor.

An image I is isotropic if it is stationary and if the covariance depends only on the spatial distance of the translations.

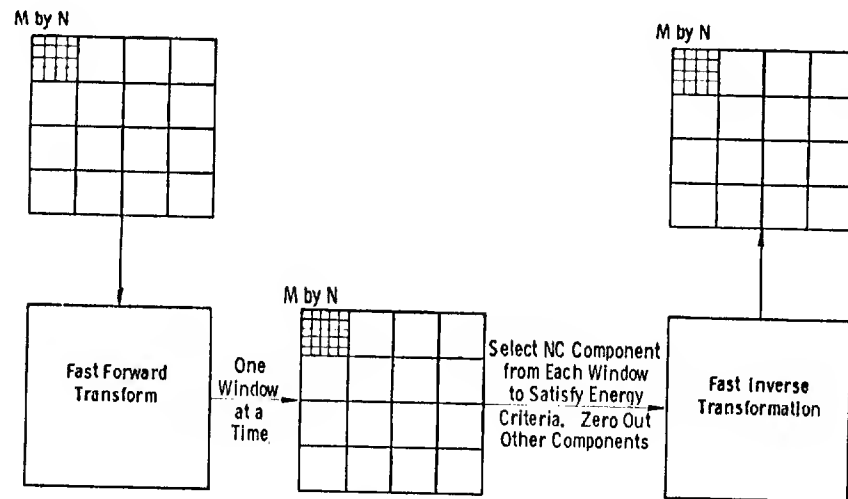


Figure 1. The transform coding technique

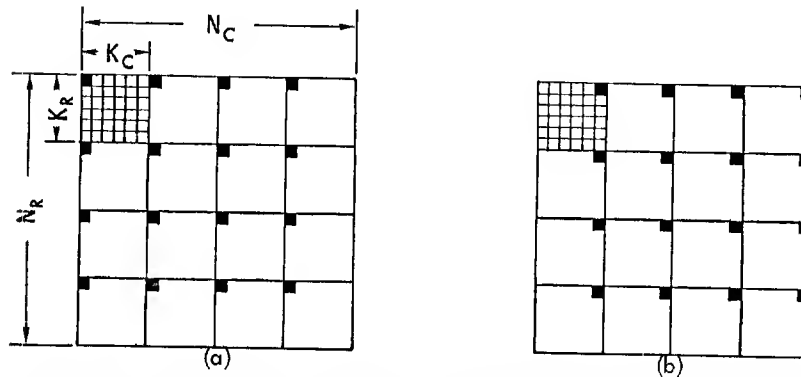


Figure 2. Illustrates that in a stationary image the average greytone taken over all resolution cells marked in image (a) equals as the average greytone taken over all resolution cells marked in image (b)

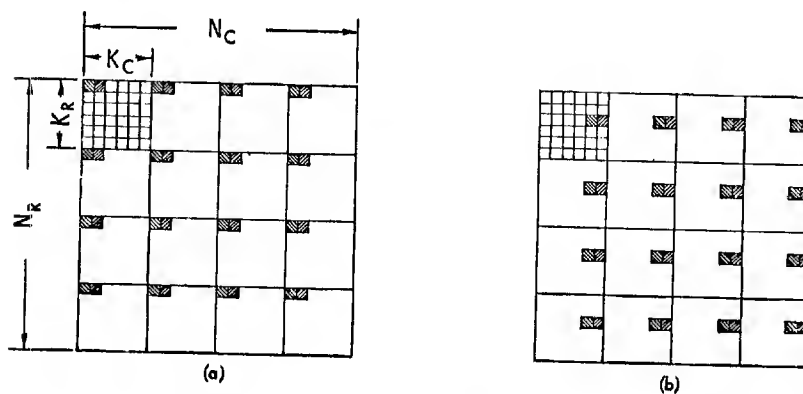


Figure 3. Illustrates that in stationary images the second moment statistics taken over all resolution cells marked in image (a) equals the second moment statistics taken over all resolution cells marked in image (b)

Thus, for example, the covariance for a shift of one row down and two columns over is not necessarily equal to the covariance for a shift of two rows down and one column over for a stationary image, but since they both represent translation of distance $\sqrt{5}$ they would be equal for an isotropic image.

In more mathematical terms, let

$Z_r = \{0, 1, \dots, N_r - 1\}$ be the set of row indexes for digital image I

$Z_c = \{0, 1, \dots, N_c - 1\}$ be the set of column indexes for digital image I.

Suppose N_c is an integer multiple of K_r and N_c is an integer multiple of K_c .

Let $R(i, j) = \{(a, b) \in Z_r \times Z_c \mid a \text{ modulo } K_r = i \text{ and } b \text{ modulo } K_c = j\}$

$S(i, j, n_1, n_2) = \{(a, b), (c, d) \mid (a, b), (c, d) \in (Z_r \times Z_c) \times (Z_r \times Z_c) \mid c = a + n_1, d = b + n_2, \text{ and } (a, b) \in R(i, j)\}$

An image $I: Z_r \times Z_c \rightarrow G$ is stationary if and only if

$$(1) \quad \mu = \frac{1}{\#R(i, j)} \sum_{(a, b) \in R(i, j)} I(a, b) \text{ for each } (i, j)$$

$$(2) \quad \sigma(n_1, n_2) = \frac{1}{\#S(i, j, n_1, n_2)} \sum_{[(a, b), (c, d)] \in S(i, j, n_1, n_2)} (I(a, b) - \mu) (I(c, d) - \mu) \text{ for each } (i, j)$$

An image $I: Z_r \times Z_c \rightarrow G$ is isotropic if and only if I is stationary and

$$\sigma(n_1, n_2) = \sigma(m_1, m_2) \text{ whenever } n_1^2 + n_2^2 = m_1^2 + m_2^2$$

Consider now how we could compute the covariance matrix for an image which satisfies or almost satisfies the stationarity or isotropicity conditions. Shown in Figure 4 is a 16x16 image. We will partition it into 4x4 subimages (Figure 5). Figure 6 shows the general form of the autocovariance matrix when no stationarity or isotropicity assumptions are made. To determine the covariance matrix with stationarity assumed, we generate a tableau which depicts all pairs of resolution cells situated in the same translational relationship [see Table 1]. From this table the stationary autocovariance may be formed, Figure 7. If the tableau is modified by the definition of isotropicity we can generate a new table and subsequent covariance array, Table 2, and Figure 8. This assumption basically establishes equivalence classes of the lettered columns in Table 1 in the following manner:

$\{a, g\}$	$\{x, r\}$
$\{q, s\}$	$\{t, l, p, j\}$
$\{v, d\}$	$\{f, n, b, h\}$
$\{w, k\}$	$\{e, u, c, o\}$
$\{m, i\}$	

Symmetry Properties of a Covariance Matrix of an Isotropic Image

A covariance matrix of an isotropic image has symmetry properties which permit the rapid computation of its eigenvectors. Such a covariance matrix can always be partitioned into K_r^2 ($K_c \times K_c$) submatrices, each submatrix being of the toeplitz form. Each such toeplitz submatrix has only K_c distinct entries as shown in Figure 9. The number of times each distinct entry occurs is given by:

entry nome	number of times entry occurs
v_1	$n_1 = K_c$
v_2	$n_2 = 2n_1 - 2$
v_3	$n_3 = n_2 - 2$
v_4	$n_4 = n_3 - 2$
\vdots	
v_{K_c-1}	$n_{K_c-1} = n_{K_c-2} - 2$
v_{K_c}	$n_{K_c} = n_{K_c-1} - 2 + 2$

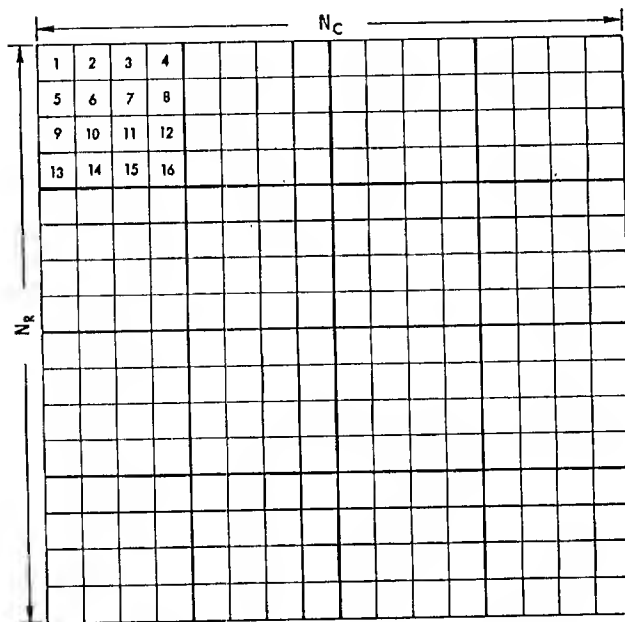


Figure 4. Illustrates a 16 x 16 image partitioned into 4 x 4 subimages.

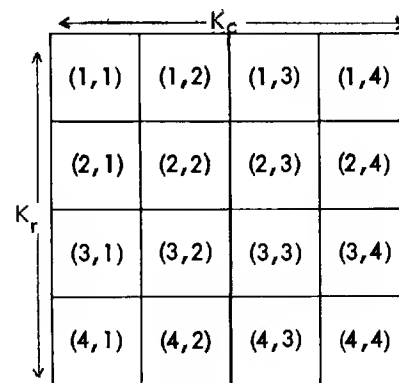


Figure 5. A $K_r \times K_c$ subimage of the original image.

Resolution Cell	(1,1)	(1,2)	(1,3)	(1,4)	(2,1)	(2,2)	(2,3)	(2,4)	(3,1)	(3,2)	(3,3)	(3,4)	(4,1)	(4,2)	(4,3)	(4,4)
(1,1)	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap
(1,2)		bb	bc	bd	be	bf	bg	bh	bi	bj	bk	bl	bm	bn	bo	bp
(1,3)			cc	cd	ce	cf	cg	ch	ci	cj	ck	cl	cm	cn	co	cp
(1,4)				dd	de	df	dg	dh	di	dj	dk	dl	dm	dn	do	dp
(2,1)					ee	ef	eg	eh	ei	ej	ek	el	em	en	eo	ep
(2,2)						ff	fg	fh	fi	fj	fk	fl	fm	fn	fo	fp
(2,3)							gg	gh	gi	gj	gk	gl	gm	gn	go	gp
(2,4)								hh	hi	hj	hk	hl	hm	hn	ho	hp
(3,1)									ii	ij	ik	il	im	in	io	ip
(3,2)										jj	jk	jl	jm	jn	jo	jp
(3,3)											kk	kl	km	kn	ko	kp
(3,4)												ll	lm	ln	lo	lp
(4,1)													mm	mn	mo	mp
(4,2)														nn	no	np
(4,3)															oo	op
(4,4)																pp

Figure 6. Illustrates that the auto-covariance matrix for a non-stationary image partitioned into 4 x 4 subimages, has 136 distinct entries. Each distinct entry is labeled with a two character label.

a <u>(3,3)</u> (1,1)-(4,4)	b <u>(3,2)</u> (1,1)-(4,3) (1,2)-(4,4)	c <u>(3,1)</u> (1,1)-(4,2) (1,2)-(4,3) (1,3)-(4,4)	d <u>(3,0)</u> (1,1)-(4,1) (1,2)-(4,2) (1,3)-(4,3) (1,4)-(4,4)	e <u>(3,-1)</u> (1,2)-(4,1) (1,3)-(4,2) (1,4)-(4,3)	f <u>(3,-2)</u> (1,3)-(4,1) (1,4)-(4,2)	g <u>(3,-3)</u> (1,4)-(4,1)
h <u>(2,3)</u> (1,1)-(3,4) (2,1)-(4,4)	i <u>(2,2)</u> (1,1)-(3,3) (1,2)-(3,4) (2,1)-(4,3) (2,2)-(4,4)	j <u>(2,1)</u> (1,1)-(3,2) (1,2)-(3,3) (1,3)-(3,4) (2,1)-(4,2) (2,2)-(4,3) (2,3)-(4,4)	k <u>(2,0)</u> (1,1)-(3,1) (1,2)-(3,2) (1,3)-(3,3) (1,4)-(3,4) (2,1)-(4,1) (2,2)-(4,2) (2,3)-(4,3) (2,4)-(4,4)	l <u>(2,-1)</u> (1,2)-(3,1) (1,3)-(3,2) (1,4)-(3,3) (2,2)-(4,1) (2,3)-(4,2) (2,4)-(4,3)	m <u>(2,-2)</u> (1,3)-(3,1) (1,4)-(3,2) (2,3)-(4,1) (2,4)-(4,2)	n <u>(2,-3)</u> (1,4)-(3,1) (2,4)-(4,1)
o <u>(1,3)</u> (1,1)-(2,4) (2,1)-(3,4) (3,1)-(4,4)	p <u>(1,2)</u> (1,1)-(2,3) (1,2)-(2,4) (2,1)-(3,3) (2,2)-(3,4) (3,1)-(4,3) (3,2)-(4,4)	q <u>(1,1)</u> (1,1)-(2,2) (1,2)-(2,3) (1,3)-(2,4) (2,1)-(3,2) (2,2)-(3,3) (2,3)-(3,4) (3,1)-(4,2) (3,2)-(4,3) (3,3)-(4,4)	r <u>(1,0)</u> (1,1)-(2,1) (1,2)-(2,2) (1,3)-(2,3) (1,4)-(2,4) (2,1)-(3,1) (2,2)-(3,2) (2,3)-(3,3) (2,4)-(3,4)	s <u>(1,-1)</u> (1,2)-(2,1) (1,3)-(2,2) (1,4)-(2,3) (2,2)-(3,1) (2,3)-(3,2) (2,4)-(3,3) (3,2)-(4,1) (3,3)-(4,2) (3,4)-(4,3)	t <u>(1,-2)</u> (1,3)-(2,1) (1,4)-(2,2) (2,3)-(3,1) (2,4)-(3,2) (3,3)-(4,1) (3,4)-(4,2)	u <u>(1,-3)</u> (1,4)-(2,1) (2,4)-(3,1) (3,4)-(4,1)
v <u>(0,3)</u> (1,1)-(1,4) (2,1)-(2,4) (3,1)-(3,4) (4,1)-(4,4)	w <u>(0,2)</u> (1,1)-(1,3) (1,2)-(1,4) (2,1)-(2,3) (2,2)-(2,4) (3,1)-(3,3) (3,2)-(3,4) (4,1)-(4,3) (4,2)-(4,4)	x <u>(0,1)</u> (1,1)-(1,2) (1,2)-(1,3) (1,3)-(1,4) (2,1)-(2,2) (2,2)-(2,3) (2,3)-(2,4) (3,1)-(3,2) (3,2)-(3,3) (3,3)-(3,4) (4,1)-(4,2) (4,2)-(4,3) (4,3)-(4,4)	y <u>(0,0)</u> (1,1)-(1,1) (1,2)-(1,2) (1,3)-(1,3) (1,4)-(1,4) (2,1)-(2,1) (2,2)-(2,2) (2,3)-(2,3) (2,4)-(2,4) (3,1)-(3,1) (3,2)-(3,2) (3,3)-(3,3) (3,4)-(3,4) (4,1)-(4,1) (4,2)-(4,2) (4,3)-(4,3) (4,4)-(4,4)			

Table 1 lists in each column all pairs of resolution cells situated in the same translational relationship. For example, all the resolution cell pairs listed under column c are related by three rows down and one row across.

Resolution Cell	(1,1)	(1,2)	(1,3)	(1,4)	(2,1)	(2,2)	(2,3)	(2,4)	(3,1)	(3,2)	(3,3)	(3,4)	(4,1)	(4,2)	(4,3)	(4,4)
(1,1)	y	x	w	v	r	q	p	o	k	j	i	h	d	c	b	a
(1,2)	x	y	x	w	s	r	q	p	l	k	j	i	e	d	c	b
(1,3)	w	x	y	x	t	s	r	q	m	l	k	j	f	e	d	c
(1,4)	v	w	x	y	u	t	s	r	n	m	l	k	g	f	e	d
(2,1)	r	s	t	u	y	x	w	v	r	q	p	o	k	j	i	h
(2,2)	q	r	s	t	x	y	x	w	s	r	q	p	l	k	j	i
(2,3)	p	q	r	s	w	x	y	x	t	s	r	q	m	l	k	j
(2,4)	o	p	q	r	v	w	x	y	u	t	s	r	n	m	l	k
(3,1)	k	l	m	n	r	s	t	u	y	x	w	v	r	q	p	o
(3,2)	j	k	l	m	q	r	s	t	x	y	x	w	s	r	q	p
(3,3)	i	j	k	l	p	q	r	s	w	x	y	x	t	s	r	q
(3,4)	h	i	j	k	o	p	q	r	v	w	x	y	u	t	s	r
(4,1)	d	e	f	g	k	l	m	n	r	s	t	u	y	x	w	v
(4,2)	c	d	e	f	j	k	l	m	q	r	s	t	x	y	x	w
(4,3)	b	c	d	e	i	j	k	l	p	q	r	s	w	x	y	x
(4,4)	a	b	c	d	h	i	j	k	o	p	q	r	v	w	x	y

Figure 7. Form of auto-covariance with stationary assumption.

$d^2=18$ $(1,1)-(4,4)$ $(1,4)-(4,1)$	$d^2=13$ $(1,1)-(4,3)$ $(1,2)-(4,4)$ $(1,3)-(4,1)$ $(1,4)-(4,2)$ $(1,4)-(3,1)$ $(2,4)-(4,1)$ $(1,1)-(3,4)$ $(2,1)-(4,4)$	$d^2=10$ $(1,1)-(4,2)$ $(1,2)-(4,3)$ $(1,3)-(4,4)$ $(1,2)-(4,1)$ $(1,3)-(4,2)$ $(1,4)-(4,3)$ $(1,4)-(2,1)$ $(2,4)-(3,1)$ $(3,4)-(4,1)$ $(1,1)-(2,4)$ $(2,1)-(3,4)$ $(3,1)-(4,4)$	$d^2=9$ $(1,1)-(4,1)$ $(1,2)-(4,2)$ $(1,3)-(4,3)$ $(1,4)-(4,4)$ $(1,1)-(1,4)$ $(2,1)-(2,4)$ $(3,1)-(3,4)$ $(4,1)-(4,4)$
	$d^2=8$ $(1,1)-(3,3)$ $(1,2)-(3,4)$ $(2,1)-(4,3)$ $(2,2)-(4,4)$ $(1,3)-(3,1)$ $(1,4)-(3,2)$ $(2,3)-(4,1)$ $(2,4)-(4,2)$	$d^2=5$ $(1,1)-(3,2)$ $(1,2)-(3,1)$ $(1,2)-(3,3)$ $(1,3)-(3,4)$ $(1,4)-(3,3)$ $(2,1)-(4,2)$ $(2,2)-(4,1)$ $(2,2)-(4,3)$ $(2,3)-(4,2)$ $(2,3)-(4,4)$ $(2,4)-(4,3)$ $(1,1)-(2,3)$ $(1,3)-(2,1)$ $(1,2)-(2,4)$ $(1,4)-(2,2)$ $(2,1)-(3,3)$ $(2,3)-(3,1)$ $(2,2)-(3,4)$ $(2,4)-(3,2)$ $(3,1)-(4,3)$ $(3,3)-(4,1)$ $(3,2)-(4,4)$ $(3,4)-(4,2)$	$d^2=4$ $(1,1)-(3,1)$ $(1,1)-(1,3)$ $(1,2)-(3,2)$ $(1,2)-(1,4)$ $(1,3)-(3,3)$ $(2,1)-(2,3)$ $(1,4)-(3,4)$ $(2,2)-(2,4)$ $(2,1)-(4,1)$ $(3,1)-(3,3)$ $(2,2)-(4,2)$ $(3,2)-(3,4)$ $(4,1)-(4,3)$ $(2,4)-(4,4)$ $(4,2)-(4,4)$
		$d^2=2$ $(1,1)-(2,2)$ $(1,2)-(2,1)$ $(1,2)-(2,3)$ $(1,3)-(2,2)$ $(1,3)-(2,4)$ $(1,4)-(2,3)$ $(2,1)-(3,2)$ $(2,2)-(3,1)$ $(2,2)-(3,3)$ $(2,3)-(3,2)$ $(2,3)-(3,4)$ $(2,4)-(3,3)$ $(3,1)-(4,2)$ $(3,2)-(4,1)$ $(3,2)-(4,3)$ $(3,3)-(4,2)$ $(3,3)-(4,4)$ $(3,4)-(4,3)$	$d^2=1$ $(1,1)-(2,1)$ $(1,1)-(1,2)$ $(1,2)-(2,2)$ $(1,2)-(1,3)$ $(1,3)-(2,3)$ $(1,3)-(1,4)$ $(1,4)-(2,4)$ $(2,1)-(2,2)$ $(2,1)-(2,3)$ $(2,2)-(3,2)$ $(2,2)-(2,4)$ $(2,3)-(3,3)$ $(3,1)-(3,2)$ $(2,4)-(3,4)$ $(3,2)-(3,3)$ $(3,1)-(4,1)$ $(3,3)-(3,4)$ $(3,2)-(4,2)$ $(4,1)-(4,2)$ $(3,3)-(4,3)$ $(4,2)-(4,3)$ $(3,4)-(4,4)$ $(4,3)-(4,4)$
			$d^2=0$ $(1,1)-(1,1)$ $(3,1)-(3,1)$ $(1,2)-(1,2)$ $(3,2)-(3,2)$ $(1,3)-(1,3)$ $(3,3)-(3,3)$ $(1,4)-(1,4)$ $(3,4)-(3,4)$ $(2,1)-(2,1)$ $(4,1)-(4,1)$ $(2,2)-(2,2)$ $(4,2)-(4,2)$ $(2,3)-(2,3)$ $(4,3)-(4,3)$ $(2,4)-(2,4)$ $(4,4)-(4,4)$

Table 2 lists in each column all pairs of resolution cells situated in the same distance relationship. For example, all the resolution cell pairs listed under column c are related by distance $=\sqrt{10}$.

Resolution Cell	(1,1)	(1,2)	(1,3)	(1,4)	(2,1)	(2,2)	(2,3)	(2,4)	(3,1)	(3,2)	(3,3)	(3,4)	(4,1)	(4,2)	(4,3)	(4,4)
(1,1)	y	r	k	d	r	q	j	c	k	j	i	b	d	c	b	a
(1,2)	r	y	r	k	q	r	q	j	j	k	j	i	c	d	c	b
(1,3)	k	r	y	r	j	q	r	q	i	j	k	j	b	c	d	c
(1,4)	d	k	r	y	c	j	q	r	b	i	j	k	a	b	c	d
(2,1)	r	q	j	c	y	r	k	d	r	q	j	c	k	j	i	b
(2,2)	q	r	q	j	r	y	r	k	q	r	q	j	j	k	j	i
(2,3)	j	q	r	q	k	r	y	r	j	q	r	q	i	j	k	j
(2,4)	c	j	q	r	d	k	r	y	c	j	q	r	b	i	j	k
(3,1)	k	j	i	b	r	q	j	c	y	r	k	d	r	q	j	c
(3,2)	j	k	j	i	q	r	q	j	r	y	r	k	q	r	q	j
(3,3)	i	j	k	j	j	q	r	q	k	r	y	r	j	q	r	q
(3,4)	b	i	j	k	c	j	q	r	d	k	r	y	c	j	q	r
(4,1)	d	c	b	a	k	j	i	b	r	q	j	c	y	r	k	d
(4,2)	c	d	c	b	j	k	j	i	q	r	q	j	r	y	r	k
(4,3)	b	c	d	c	i	j	k	j	j	q	r	q	k	r	y	r
(4,4)	a	b	c	d	b	i	j	k	c	j	q	r	d	k	r	y

Figure 8. Isotropic auto-covariance form

The total number of distinct submatrices in the partition of the covariance matrix is K_r . The submatrices themselves are organized as a toeplitz form. As shown in Figure 10 the number of times each submatrix is repeated in the covariance matrix is given by:

submatrix	number of times submatrix occurs
S_1	$m_1 = K$
S_2	$m_2 = 2m_1 - 2$
S_3	$m_3 = m_2 - 2$
\vdots	
S_{K_r-1}	$m_{K_r-1} = m_{K_r-2} - 2$
S_{K_r}	$m_{K_r} = m_{K_r-1} - 2 = 2$

	K_c				
K_c	v_1	v_2	v_3	v_4	v_5
	v_2	v_1	v_2	v_3	v_4
	v_3	v_2	v_1	v_2	v_3
	v_4	v_3	v_2	v_1	v_2
	v_5	v_4	v_3	v_2	v_1

Figure 9. Illustrates a 5x5 submatrix of the covariance matrix of an isotropic image. The submatrix is toeplitz and has only five distinct entries: v_1, v_2, v_3, v_4, v_5 .

S_1	S_2	S_3	S_4	S_5
S_2	S_1	S_2	S_3	S_4
S_3	S_2	S_1	S_2	S_3
S_4	S_3	S_2	S_1	S_2
S_5	S_4	S_3	S_2	S_1

Figure 10. Illustrates how the covariance matrix of an isotropic image can be partitioned into submatrices each of the toeplitz form shown in Figure 9.

Figure 11 shows the form of a covariance matrix for an isotropic image partitioned into 3 row by 4 column subimages. With all this symmetry, we certainly should expect that the calculation of the eigenvectors of this matrix involves less work than we would need to do for a general covariance matrix. Perhaps if we are lucky, the transformation defined by the eigenvectors may even have a fast implementation.

i g f	g h e	f e d	c b a
g i g	h g h	e f e	b c b
f g i	e h g	d e f	a b c
g h e	i g f	g h e	f e d
h g h	g i g	h g h	e f e
e h g	f g i	e h g	d e f
f e d	g h e	i g f	g h e
e f e	h g h	g i g	h g h
d e f	e h g	f g i	e h g
c b a	f e d	g h e	i g f
b c b	e f e	h g h	g i g
a b c	d e f	e h g	f g i

= \mathcal{F}

Figure 11. Isotropic Auto-covariance form when an Isotropic Image is partitioned into 3 x 4 subimages $K_c = 3$
 $K_r = 4$

Theory of Composite Matrices

Our situation is a fortunate one because the theory of composite matrices says that, if a matrix A can be partitioned into submatrices which have the same set of eigenvectors, then the eigenvectors of A can be formed as the direct product of the eigenvectors of the submatrices with the eigenvectors of the matrix of corresponding eigenvalues. Hence if:

$$A = \begin{pmatrix} A_1 & A_2 & A_3 \\ A_4 & A_5 & A_6 \\ A_7 & A_8 & A_9 \end{pmatrix}$$

and v is on eigenvector of each submatrix A_i with corresponding eigenvalue λ_i , and u is on eigenvector of the matrix

$$\begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 \\ \lambda_4 & \lambda_5 & \lambda_6 \\ \lambda_7 & \lambda_8 & \lambda_9 \end{pmatrix}$$

with corresponding eigenvalue η then: the direct product* uv is an eigenvector of A with corresponding eigenvalue η . The covariance matrix for the isotropic image almost has the required property of the composite matrix. The covariance matrix can be partitioned into submatrices all of the same toeplitz form. However, this is not a guarantee that the submatrices have the same eigenvectors.

Empirical work with the submatrices of the covariance matrix of an isotropic image indicates that the submatrices are almost multiples of one another and therefore have almost the same eigenvector set. This justifies the following heuristic. Generate a submatrix with the property that the normed squares of the matrix of the differences between the best multiple of it and any given submatrix of the covariance matrix is the smallest when averaged over all submatrices. Then replace each submatrix with the best multiple of the generated submatrix. This creates a covariance matrix having the required composite structure. The eigenvectors can be determined quickly and the transformation defined by the eigenvectors has a fast implementation.

To demonstrate this approximation, consider the Isotropic Auto-covariance matrix in Figure 11. If we generate the second moment matrix considering each submatrix as a vector the following form emerges:

$$\begin{array}{c} \begin{array}{c} \uparrow \\ K_c \\ \downarrow \end{array} \begin{array}{c} \leftarrow K_c \rightarrow \\ \begin{array}{|c|c|c|} \hline p & q & r \\ \hline q & t & s \\ \hline r & s & u \\ \hline \end{array} \end{array} \begin{array}{c} \begin{array}{|c|c|c|} \hline q & p & q \\ \hline t & q & t \\ \hline s & r & s \\ \hline \end{array} \end{array} \begin{array}{c} \begin{array}{|c|c|c|} \hline r & q & p \\ \hline s & t & q \\ \hline u & s & r \\ \hline \end{array} \end{array} \end{array} = \Phi$$

Figure 12. Form of the second moment matrix derived from the isotropic covariance matrix with each submatrix as a vector. All entries labeled q , for example, are the sums of products of entries labeled i and g respectively of the matrix in Figure 11.

Notice that all the distinct variables are in the first submatrix. It is, therefore, possible to find the eigenvector having largest eigenvalue of this matrix without forming the entire matrix. All that is necessary is to store the first submatrix of the second moment matrix.

Let $v^1 = (v_1, \dots, v_9)$ be the eigenvector of Φ_1 having largest eigenvalue λ . Then $\Phi_1 v = \lambda v$. The first row of the matrix equation is

$$pv_1 + qv_2 + rv_3 + qv_4 + pv_5 + qv_6 + rv_7 + qv_8 + pv_9 = \lambda v_1$$

and the fifth row is

$$pv_1 + qv_2 + rv_3 + qv_4 + pv_5 + qv_6 + rv_7 + qv_8 + pv_9 = \lambda v_5$$

Obviously, the first row and the fifth row are equal so $v_5 = v_1$. In like manner, the other rows may be compared, resulting in:

$$\begin{cases} v_1 = v_5 = v_9 \\ v_2 = v_4 = v_6 = v_8 \\ v_3 = v_7 \end{cases} \quad \begin{cases} 3pv_1 + 4qv_2 + 2rv_3 = \lambda v_1 \\ 3qv_1 + 4tv_2 + 2sv_3 = \lambda v_2 \\ 3rv_1 + 4sv_2 + 2uv_3 = \lambda v_3 \end{cases}$$

Separating these simultaneous equations we have

$$\begin{bmatrix} 3p & 4q & 2r \\ 3q & 4t & 2s \\ 3r & 4s & 2u \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \lambda \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

This always happens because in the general case we have only K_c independent equations, all of which make up the first submatrix, of the second moment matrix, derived from an isotropic covariance array in this manner. We need not worry about which elements are equal as long as our index follows the toeplitz form. By using the resulting eigenvector and symmetry the appropriate multiples for each submatrix of the auto-covariance matrix can be determined. The coefficients found by this procedure may be illustrated by: $c_1 = (v_1 \ v_2 \ v_3 \ v_2 \ v_1 \ v_2 \ v_3 \ v_2 \ v_1) \ (i \ g \ f \ g \ i \ g \ f \ g \ i)^t$

* If $u^1 = (u_1, \dots, u_n)$ and $v^1 = (v_1, \dots, v_m)$, then the direct product $(uv)^1 = (u_1 v_1, u_1 v_2, \dots, u_1 v_m, u_2 v_1, u_2 v_2, \dots, u_2 v_m, \dots, u_n v_1, u_n v_2, \dots, u_n v_m)$

or as $c_1 = (v_1 \ v_2 \ v_3) \begin{pmatrix} 3i \\ 4g \\ 2f \end{pmatrix}$ using just the first three elements.

Likewise:

$$c_2 = (v_1 \ v_2 \ v_3) \begin{pmatrix} 3g \\ 4h \\ 2e \end{pmatrix}$$

$$c_3 = (v_1 \ v_2 \ v_3) \begin{pmatrix} 3f \\ 4e \\ 2d \end{pmatrix}$$

$$c_4 = (v_1 \ v_2 \ v_3) \begin{pmatrix} 3c \\ 4b \\ 2a \end{pmatrix}$$

The composite matrix Σ_c can then be immediately formed as:

$$\Sigma_c = \begin{bmatrix} c_1 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_3 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_4 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} \\ c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_1 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_3 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} \\ c_3 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_1 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} \\ c_4 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_3 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_2 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} & c_1 \begin{bmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{bmatrix} \end{bmatrix}$$

The composite matrix resulting from this operation has several properties which should be recognized: First the submatrices of the composite matrix all have the same eigenvectors (they commute). Second, each submatrix is of the same toeplitz form. Third, corresponding to the shared eigenvectors V_i is the lambda matrix of corresponding eigenvalues,

$$\begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 \\ \lambda_5 & \lambda_6 & \lambda_7 & \lambda_8 \\ \lambda_9 & \lambda_{10} & \lambda_{11} & \lambda_{12} \\ \lambda_{13} & \lambda_{14} & \lambda_{15} & \lambda_{16} \end{pmatrix}, \quad \lambda_i \text{ being the eigenvalue of } c_i \begin{pmatrix} v_1 & v_2 & v_3 \\ v_2 & v_1 & v_2 \\ v_3 & v_2 & v_1 \end{pmatrix}$$

under the eigenvector V_i . Obviously, for each V_i the lambda matrices differ only by a multiplicative constant and so also share the same eigenvectors. The direct product of these two sets of eigenvectors are the eigenvectors of Σ_c . Since the eigenvectors of Σ_c can be represented by the direct product of two sets of vectors, the eigenvector transformation has a fast implementation. (See Figures 13 and 14).

Once the fast implementation has been defined by the direct product relation of the shared eigenvectors of the submatrices and the shared eigenvectors of the lambda matrices we have created a set of basis vectors which may be used to define an orthogonal transformation on an image. (See Figures 15 and 16).

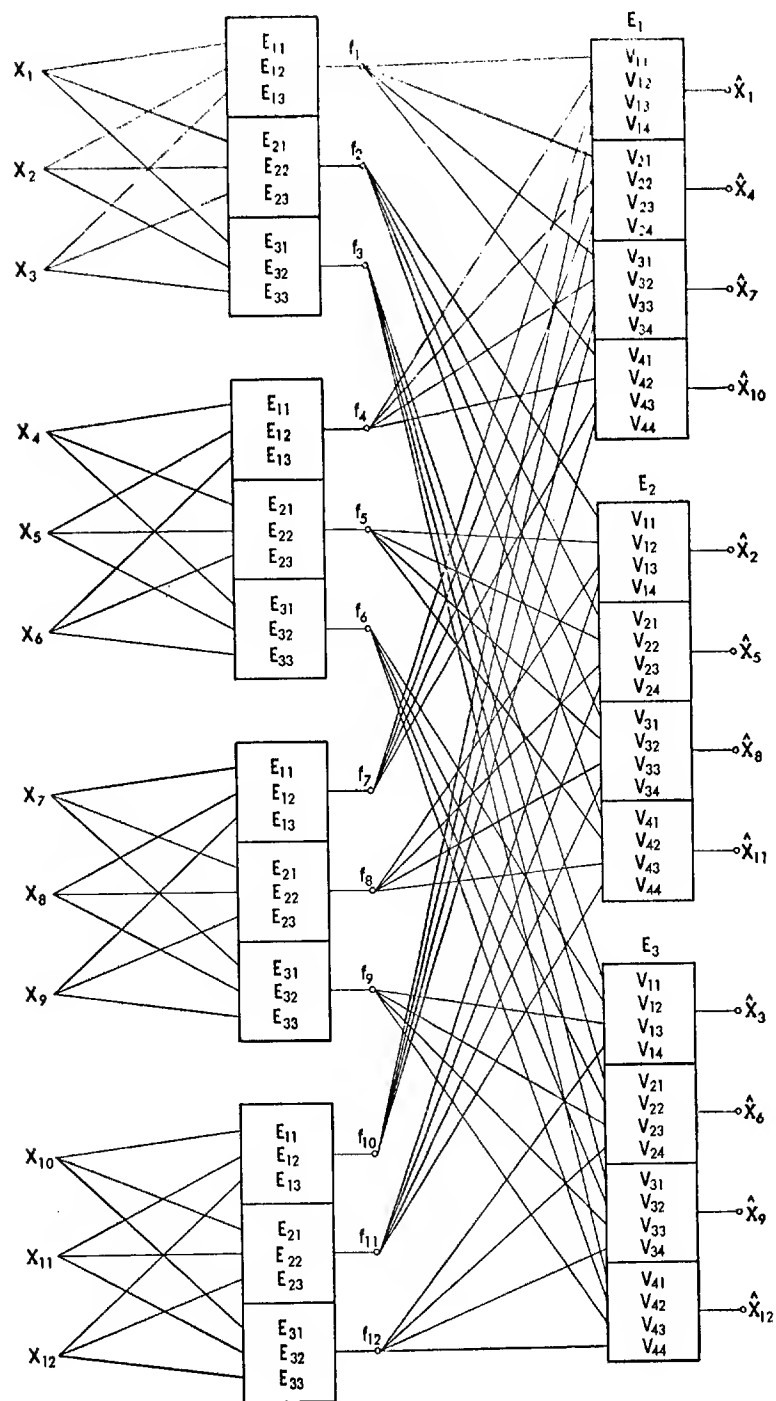


Figure 13. Each subimage has three rows by four columns. The fast implementation transforms each of the four rows of the subimage first and then transforms each of the three resulting columns.

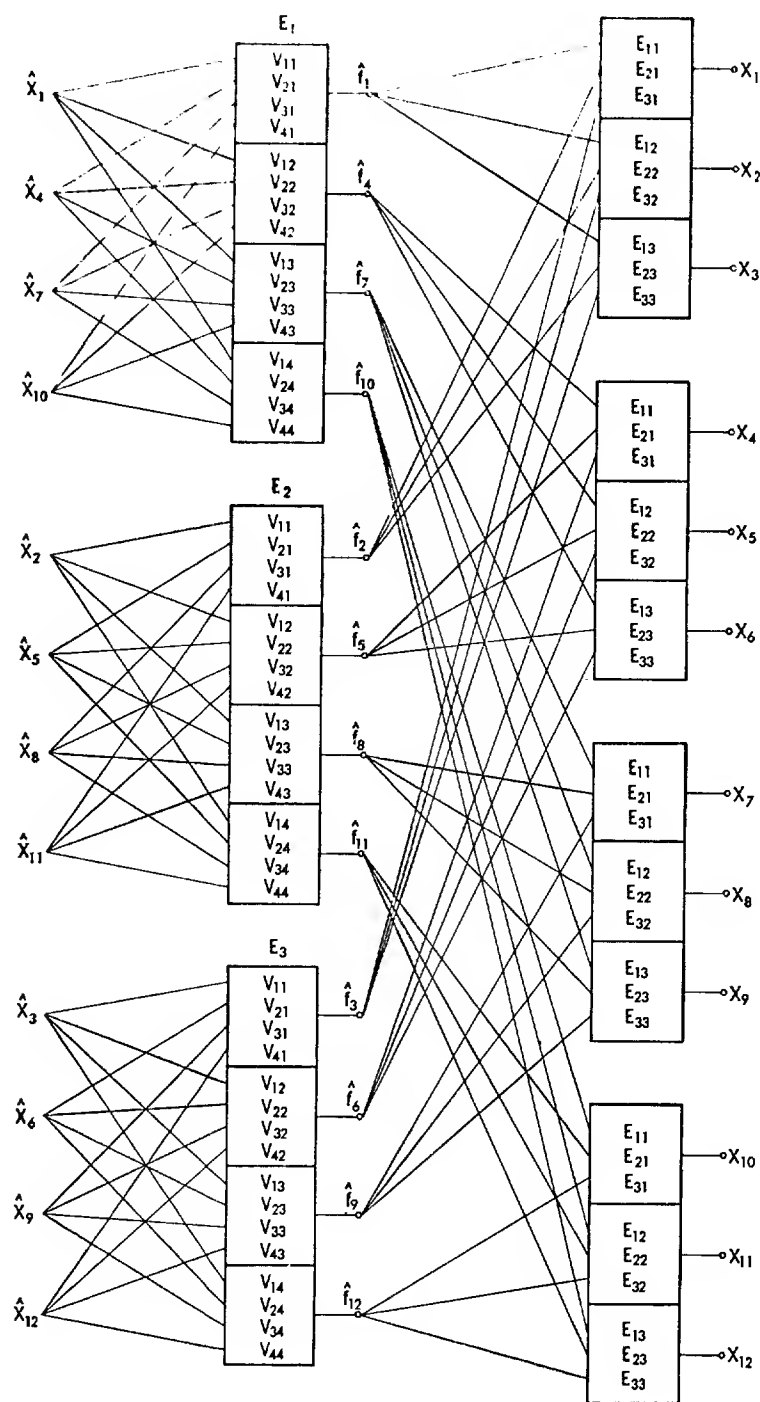


Figure 14. The inverse transform operates on each of the three columns and then operates on each of the four resulting rows.

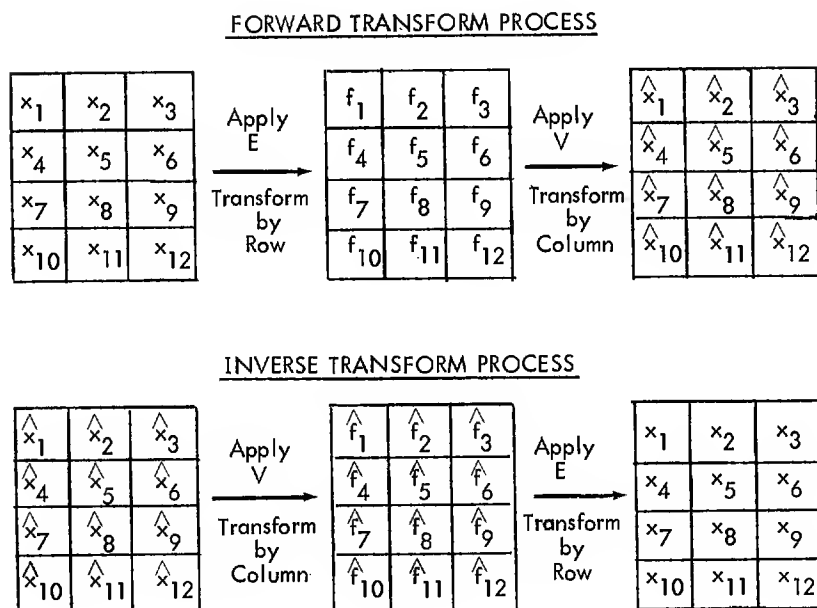


Figure 15. Row-Column operation with basis vectors.

$$E = \left\{ \begin{pmatrix} E_{11} \\ E_{12} \\ E_{13} \end{pmatrix}, \begin{pmatrix} E_{21} \\ E_{22} \\ E_{23} \end{pmatrix}, \begin{pmatrix} E_{31} \\ E_{32} \\ E_{33} \end{pmatrix} \right\} \text{ orthogonal row transformation basis vectors}$$

$$V = \left\{ \begin{pmatrix} v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \end{pmatrix}, \begin{pmatrix} v_{21} \\ v_{22} \\ v_{23} \\ v_{24} \end{pmatrix}, \begin{pmatrix} v_{31} \\ v_{32} \\ v_{33} \\ v_{34} \end{pmatrix}, \begin{pmatrix} v_{41} \\ v_{42} \\ v_{43} \\ v_{44} \end{pmatrix} \right\} \text{ orthogonal column transformation basis vectors}$$

Figure 16. Basis vector sets derived from composite matrix for direct product implementation.

Computational Results

In terms relative to the transform coding of the image, what we have essentially done is to see under what conditions the Karhunen-Loeve transform is separable so that each subimage can be transformed by first operating on its rows and then by columns. We determined that in order for this to happen the image had to be isotropic and we had to approximate the submatrices of the isotropic covariance matrix with submatrices having the same eigenvectors. In order to determine if the image isotropicity assumption and the approximations were good ones for image data we ran some experiments comparing the standard K-L transform with the fast approximate K-L transform. We found, as described in the following discussion, that the fast approximate K-L transform gives results almost as good as the Hadamard, Fourier, and Slant. A 512 by 512, 6 bit digital image was compressed at ratios of 2 bits/pel, 1 bit/pel, and .5 bits/pel. A comparison was made between the Standard K-L transform, the Fast K-L transform, the Discrete Linear Basis, and the SLANT transform. The error criteria chosen was the RMS and RMS correlated measures (Haralick & Shanmugam, 1974).

The first error investigated was that of the stationarity and isotropic assumptions. The L₂ matrix norm of the differences between the various combinations are:

$$\begin{aligned} \| \text{Auto-Covariance} - \text{Stationary Covariance} \| &= 5.7628 \\ \| \text{Stationary Covariance} - \text{Isotropic Covariance} \| &= 5.5923 \\ \| \text{Auto-Covariance} - \text{Isotropic Covariance} \| &= 4.5410 \end{aligned}$$

These RMS errors should be compared to 401.765, the average value of the diagonal elements of the covariance matrices. If we let A, S, I represent the auto-covariance, of the original image, the stationary covariance, and isotropic covariance respectively, then:

$$(A-I) = (A-S) - (I-S)$$

$$\|A-I\| \leq \|A-S\| + \|I-S\|$$

and our results satisfy the triangular inequality. In terms of distance measure Figure 17 depicts the differences of these approximations. The percentage difference, from the average variance of the auto-covariance, is 1.434% for the stationary assumption and 1.130% for the isotropic assumption. Examples of the actual auto-covariance matrix and auto-covariance matrices under the stationarity and isotropicity assumption are included in Figures 18, 19, and 20 respectively.

The eigenvectors of the standard auto-covariance matrix do not, in general, have the discrete sequence property which other fast transforms have. The eigenvectors resulting from the fast K-L do have the sequence property as the other fast transforms have. The eigenvectors for both the standard K-L and Fast K-L are compared in Figures 21 and 22. The subimage size used was 4 x 4 only because of memory and computational restrictions on determining the standard K-L. The fast version can use much larger subimages because of the symmetry and storage savings.

Results of the compression at 2 bits/pel, 1 bit/pel, and .5 bits/pel are plotted for comparison in Figure 23. It is apparent that the Fast K-L outperforms the other transforms and is closest to the optimum, differing from the optimum KL only by approximately 1%. Table 3 gives a comparison of these errors.

Conclusions

In terms relative to the transform coding of an image we have presented the conditions under which the Karhunen-Loeve transform is separable and developed an approximate fast K-L transform. We have discussed the approximations of stationarity and isotropicity in terms of the L₂ distance norms. Our results indicate that the fast K-L transform is comparable to other discrete transforms both in terms of sequence and compression performance. Experimental data indicates that the fast K-L transform is closest to the optimum, differing from the optimum K-L by approximately 1%.

References

- S. N. Afrait, "Composite Matrices," *Quarterly Journal of Mathematics*, Oxford, Vol. 2, No. 5, 1954, pp. 81-88.
- I. J. Good, "The Relationship Between Two Fast Fourier Transforms," *IEEE C-20*, March, 1971, pp. 310-317.
- R. M. Haralick and K. Shanmugam, "Comparative Study of a Discrete Linear Basis for Image Data Compression," *IEEE SMC-4*, No. 1, Jan., 1974, pp. 16-27.
- R. M. Haralick and N. Griswold, "Image Data Compression: The Incomplete Fast Transform," *IEEE SMC-CHO 908-4*, Oct., 2-4, 1974, pp. 299-306.
- K. Shanmugam and R. M. Haralick, "A Computationally Simple Procedure for Imagery Data Compression by the Karhunen-Loeve Method," *IEEE SMC-3*, No. 2, March, 1973, pp. 202-204.
- J. Williamson, "The Latent Roots of a Matrix of Special Type," *Bulletin Amer. Math. Society*, Vol. 37, 1931, pp. 585-590.

Acknowledgment

This work was supported in part by the Air Force Avionics Laboratory, Wright-Patterson Air Force Base Ohio, U. S. Air Force contract number F33615-74-C-1093.

R. M. Haralick is with the Remote Sensing Lab and the Department of Electrical Engineering, University of Kansas, Lawrence, Kansas, 66045.

N. Griswold is with the Air Force Avionics Laboratory presently working at the Remote Sensing Lab, University of Kansas, Lawrence, Kansas, 66045.

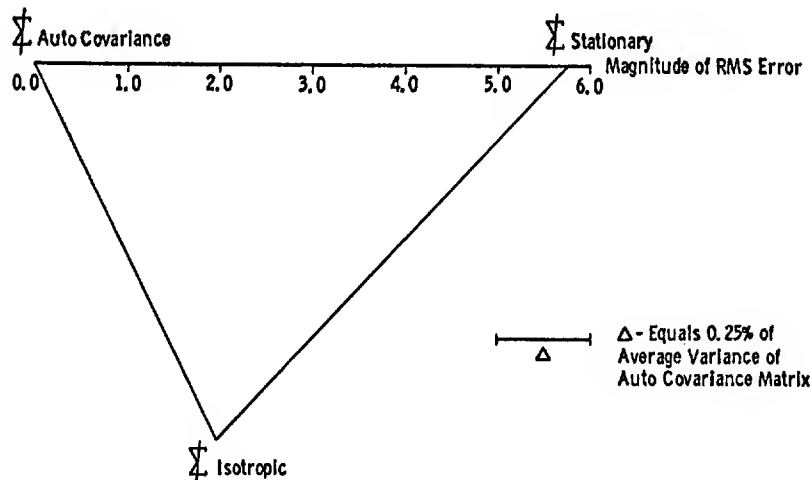


Figure 17. Illustrates the distance between the actual auto-covariance matrix computed with no assumption, the stationarity assumption and the isotropic assumption.

401.757	383.327	370.692	361.734	379.608	372.763	364.253	356.036	365.537	361.459	354.369	347.887	356.392	352.230	346.560	341.021
	401.384	383.142	370.850	373.542	379.508	372.179	363.446	361.543	365.361	360.092	353.700	353.119	356.311	352.258	345.568
		399.991	382.111	364.907	373.654	378.506	371.421	355.281	361.578	363.734	359.484	347.753	353.029	355.857	350.502
			398.535	357.480	364.666	373.308	377.281	349.812	355.731	360.737	362.931	343.394	348.069	352.996	353.300
				401.457	384.079	372.026	362.976	380.793	374.374	365.442	358.569	367.994	363.172	356.967	350.775
					401.232	383.675	371.883	375.629	380.965	373.366	365.432	364.721	367.873	363.196	356.167
						400.745	383.357	366.884	375.817	379.598	373.212	358.967	364.989	367.346	362.053
							399.955	359.253	366.115	374.280	379.014	352.356	358.484	364.376	365.338
								403.803	387.230	374.571	366.422	383.538	376.528	368.225	360.772
									404.067	385.970	374.307	378.707	383.698	376.457	367.586
										402.801	385.748	368.920	377.935	383.012	375.718
											402.767	361.521	369.193	379.023	381.941
												403.088	385.209	373.348	364.143
													403.017	385.181	373.437
														402.689	385.654
															400.782

Figure 18. Auto-covariance matrix derived from original image (upper triangle).

401.801	384.426	372.673	363.811	380.581	373.757	365.495	357.971	366.498	362.080	355.877	349.747	356.304	352.293	346.786	341.426
	401.801	384.426	372.673	380.556	380.581	373.757	365.495	366.471	366.498	362.080	355.877	356.273	356.304	352.293	346.786
		401.801	384.426	380.543	380.556	380.581	373.757	366.455	366.471	366.498	362.080	356.251	356.273	356.304	352.293
			401.801	380.518	380.543	380.556	380.581	366.425	366.455	366.471	366.498	356.227	356.251	356.273	356.304
				401.801	384.426	372.673	363.811	380.581	373.757	365.495	357.971	366.498	362.080	355.877	349.747
					401.801	384.426	372.673	380.556	380.581	373.757	365.495	366.471	366.498	362.080	355.877
						401.801	384.426	380.543	380.556	380.581	373.757	366.455	366.471	366.498	362.080
							401.801	380.518	380.543	380.556	380.581	366.425	366.455	366.471	366.498
								401.801	384.426	372.673	363.811	380.581	373.757	365.495	357.971
									401.801	384.426	372.673	380.556	380.581	373.757	365.495
										401.801	384.426	380.543	380.556	380.581	373.757
											401.801	380.518	380.543	380.556	380.581
												401.801	384.426	372.673	363.811
													401.801	384.426	372.673
														401.801	384.426
															401.801

Figure 19. Auto-covariance matrix derived using stationary assumption.

401.801	382.504	369.586	360.058	382.504	377.156	368.647	361.764	369.586	368.647	361.166	354.802	360.058	361.764	354.802	348.827
	401.801	382.504	369.586	377.156	382.504	377.156	368.647	368.647	369.586	368.647	361.166	361.764	360.058	361.764	354.802
		401.801	382.504	368.647	377.156	382.504	377.156	361.166	368.647	369.586	368.647	354.802	361.764	360.058	361.764
			401.801	361.764	368.647	377.156	382.504	354.802	361.166	368.647	369.586	348.827	354.802	361.764	360.058
				401.801	382.504	369.586	360.058	382.504	377.156	368.647	361.764	369.586	368.647	361.166	354.802
					401.801	382.504	369.586	377.156	382.504	377.156	369.647	368.647	369.586	368.647	361.166
						401.801	382.504	368.647	377.156	382.504	377.156	361.166	368.647	369.586	368.647
							401.801	361.764	368.647	377.156	382.504	354.802	361.166	368.647	369.586
								401.801	382.504	369.586	360.058	382.504	377.156	368.647	361.764
									401.801	382.504	369.586	377.156	382.504	377.156	368.647
										401.801	382.504	368.647	377.156	382.504	377.156
											401.801	361.764	368.647	377.156	382.504
												401.801	382.504	369.586	360.058
													401.801	382.504	369.586
														401.801	382.504
															401.801

Figure 20. Auto-covariance matrix derived using isotropic assumption.

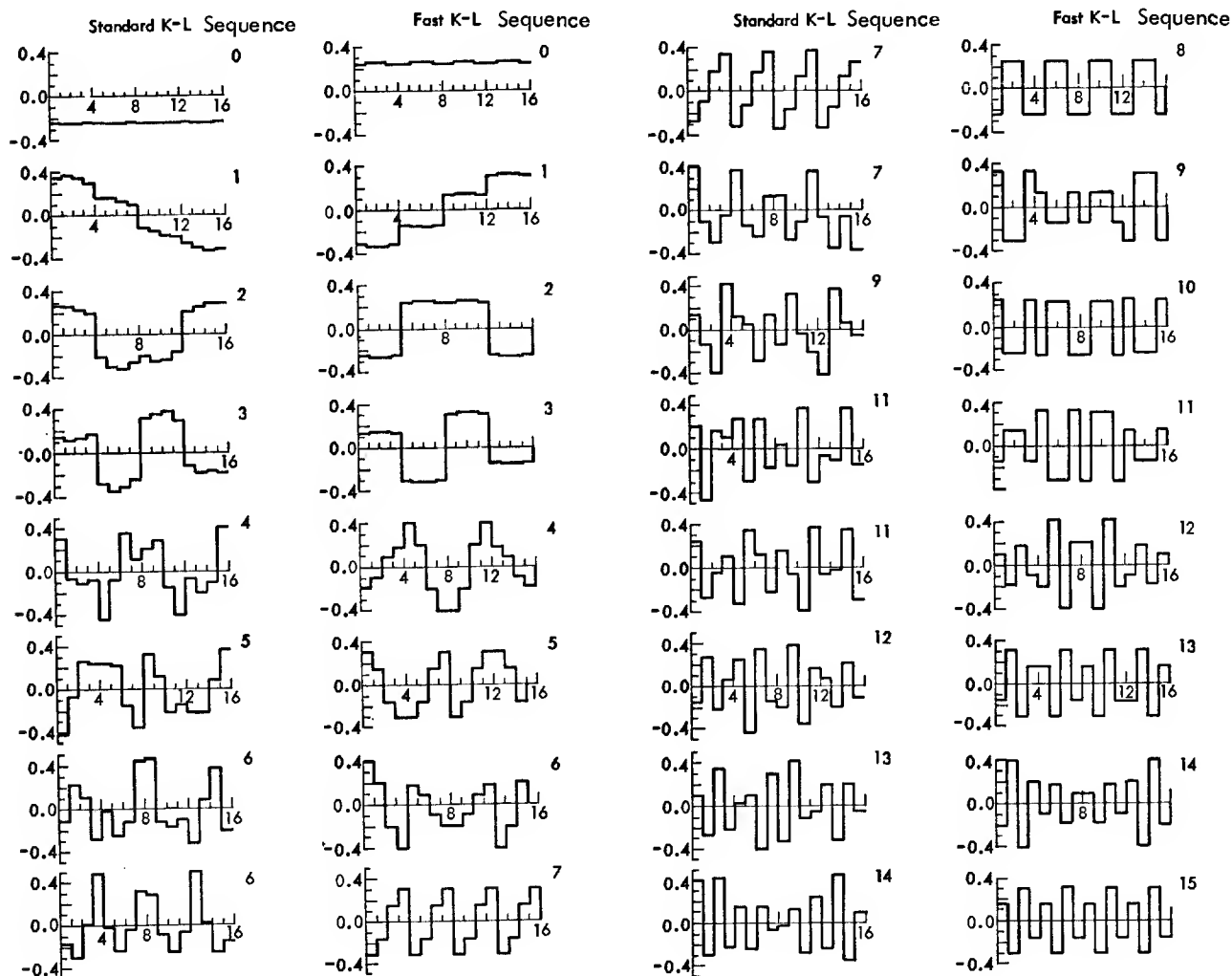


Figure 21-22. Comparison of eigenvectors for Standard K-L and Fast K-L in order of sequence.

TABLE 3

TABLE OF RMS ERROR

TRANSFORM	COMPRESSION RATIO		
	2.0 bits/pel	1.0 bits/pel	.5 bits/pel
STANDARD K-L	2.9071	4.1855	4.9461
FAST K-L	2.9362	4.2098	5.0094
DLB	2.9492	4.2819	5.1993
SLANT	2.9492	4.2819	5.1993

% ERROR COMPARED TO STANDARD K-L

TRANSFORM	COMPRESSION RATIO		
	2.0 bits/pel	1.0 bits/pel	.5 bits/pel
FAST K-L	1.000	.580	1.279
DLB	1.448	2.303	5.119
SLANT	1.448	2.303	5.119

RMS CORRELATED ERROR BETWEEN STANDARD K-L AND FAST K-L

COMPRESSION RATIO

	DISTANCE	2.0 bits/pel		1.0 bits/pel		.5 bits/pel	
		RMS ERROR	CORREL. ERROR	RMS ERROR	CORREL. ERROR	RMS ERROR	CORREL. ERROR
STANDARD K-L	1	2.9071	.10238	.2976	4.1855	.12813	.5363
	2	2.9071	.10202	.2966	4.1855	.12653	.5296
	5	2.9071	.09966	.2897	4.1855	.12400	.5140
	10	2.9071	.09400	.2733	4.1855	.12036	.5038
FAST K-L	1	2.9362	.10415	.3058	4.2098	.10424	.4388
	2	2.9362	.10212	.2998	4.2098	.10509	.4424
	5	2.9362	.10129	.2974	4.2098	.10862	.4572
	10	2.9362	.09654	.2835	4.2098	.10613	.4468

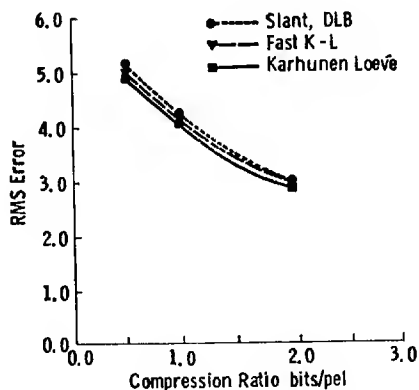


Figure 23a. Comparison of RMS Error as a function of compression ratio for K-L, Fast K-L, Slant, and DLB.

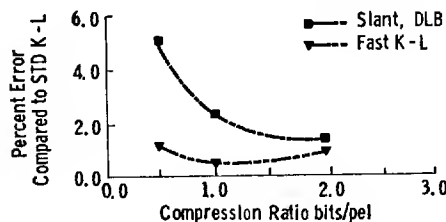


Figure 23b. Comparison of percentage error as a function of compression ratio for Fast K-L, Slant, and DLB.

Singular Value Decomposition for Image Coding*

Harry C. Andrews
University of Southern California
Image Processing Institute
Los Angeles, California 90007

Abstract

The numerical algorithm known as singular value decomposition [1] has been applied to image processing with interesting consequences [2]. Initially the algorithm was used to effect a pseudoinverse restoration [3] for better object estimation from degraded imagery. However the singular value decomposition (SVD) technique can also be effectively utilized in image compression problems when large compression ratios are desired. This Abstract refers to some experimental work developed at The Aerospace Corporation in which imagery has been compressed using SVD methods. Pictorial and computational results indicate that 8 bit imagery can be compressed to 2 bits with 0.5% mean square error while the same imagery can be compressed to 1/2 bit with 1.5% mean square error. These results will be presented along with various algorithms which make use of the SVD domain of an image for subsequent image compaction.

References

1. Golub, G. H., and C. Reinsch, "Singular Value Decomposition and Least Squares Solutions," Numer. Math., Vol. 14 p. 403-420 (1970)
2. Andrews, H. A. and C. L. Patterson, "Outer Product Expansions and Their Uses in Digital Image Processing" Am. Math Monthly, Vol. 1, No. 682, p. 1-13 (1975)
3. Albert, A. Regression and the Moore-Penrose Pseudoinverse, Academic Press, New York, New York, (1972)

* Portions of this research were sponsored by the Advanced Research Projects Agency of the Department of Defense under contract number F08606-72-C-0008 as well as by internal fee sponsored research funds within The Aerospace Corporation.

TV BANDWIDTH COMPRESSION

Paul Wintz*
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

Jim Wilson
WINTEK Corp.
902 N. 9th St.
Lafayette, Indiana 47904

Ernie Schmidt
Aeronutronic Ford Corp.
Willow Grove, Pennsylvania 19090

There are three possibilities for reducing the bandwidth of a Television signal: (1) reduce the temporal resolution by transmitting less than 30 frames per second; (2) reduce the spatial resolution by coding fewer than all of the picture elements in the frames transmitted; (3) reduce the spatial redundancy in the remaining picture elements.

The purpose of this project is to investigate the tradeoffs between the latter two methods for reducing the number of bits required to code the picture.

The original picture was digitized into a 512x512 array of picture elements of 8 bits each. The resolution was then reduced to 256x256 picture elements by averaging subsets of 2x2 picture elements in the 512x512 picture. The 512x512 picture is presented in Fig. 1 and the 256x256 picture is presented in Fig. 5.

Each of these pictures was processed by the hybrid coding scheme due to Habibi⁽¹⁾. Each line of picture elements was first divided into sub-blocks containing 32 picture elements each. Each of these 1x32 arrays of picture elements was transformed with the cosine transformation to obtain 32 coefficients. The cosine transformation was performed on each 1x32 subsets of the entire original picture in order to reduce the redundancy in the horizontal direction. Successive scan lines are also highly correlated. Since the picture elements in a 1x32 sub-array of picture elements are highly correlated with the picture elements in the 1x32 sub-array on the next scan line, and since we do the same transformation on both sub-arrays, we also expect the coefficients produced by the transformations to be highly correlated. This redundancy can be reduced by differencing the coefficients. This can be accomplished by the coding scheme known as DPCM. In summary, we use the cosine transform in the horizontal direction followed by DPCM in the vertical direction to obtain 1x32 arrays of the coefficient differences.

Since the coefficients have different variances it follows that the differences between coefficients will also have different variances. In order to efficiently code the coefficient differences we use block quantization. That is, we use a different number of bits to code the different coefficient differences. The bit assignment is optimized in order to minimize the RMS quantization error. The bit allocations are listed in Table 1.

The picture in Fig. 1 was coded by this method and then decoded to form a 512x512 replica of the original image. The decoded pictures for 2, 1, and 1/2 bits per picture element are presented in Figs. 2, 3, and 4. The 2-bit picture has only a very small degradation relative to the original. The one-bit picture has more degradation. The 1/2-bit picture not only has very noticeable degradation but the edges of the 1x32 blocks are also apparent.

The same coding scheme was applied to the 256x256 picture of Fig. 5. The coded pictures were decoded to form a replica of the 256x256 image which was then converted to a 512x512 image by repeating each picture element. The results are shown in Figs. 6, 7, and 8 for 2, 1, and 1/2 bits per picture element. The same general comments apply as for the 512x512 picture except that the distortions are significantly more severe. The reason for this is that there is less correlation between the 256x256 picture elements than between the 512x512 picture elements and so the coding algorithm is expected to be less efficient.

Finally, we compare the 512x512 picture coded at 1/2-bit per picture element illustrated in Fig. 4 with the 256x256 picture coded at 2 bits per picture element shown in Fig. 8. Both of these pictures require the same number of bits to transmit. The 512x512 picture at 1/2 bit is subjectively better than the 256x256 picture at 2 bits.

The same experiment was performed on a different picture and the decoded pictures are presented in Figs. 9 to 16. Fig. 9 is the 512x512 original, and Figs. 10, 11, and 12 are coded at 2 bits, 1 bit, and 1/2 bit respectively. Fig. 13 is the 256x256 original and Figs. 14, 15, and 16 correspond to coding this picture with 2 bits, 1 bit, and 1/2 bit. Careful observation of these pictures leads to the same conclusion as for the previous pictures.

*Paul Wintz, President, WINTEK Corp., 902 N. 9th St., Lafayette, Indiana 47904.

Table 1 Bit Allocation for Coefficient Differences

<u>Coefficients</u>	<u>2 bits</u>		<u>1 bit</u>		<u>1/2 bit</u>	
	<u>512 picture</u>	<u>256 picture</u>	<u>512 picture</u>	<u>256 picture</u>	<u>512 picture</u>	<u>256 picture</u>
1	4	4	3	3	3	3
2	4	3	3	2	2	2
3	3	3	2	2	2	2
4	3	3	2	2	1	1
5	3	3	2	2	1	1
6	3	3	2	2	1	1
7	3	3	2	2	1	1
8	3	3	2	2	1	1
9	3	3	2	2	1	1
10	3	2	2	1	1	1
11	3	2	2	1	1	1
12	2	2	1	1	1	1
13	2	2	1	1	0	0
14	2	2	1	1	0	0
15	2	2	1	1	0	0
16	2	2	1	1	0	0
17	2	2	1	1	0	0
18	2	2	1	1	0	0
19	2	2	1	1	0	0
20	1	2	0	1	0	0
21	1	2	0	1	0	0
22	1	2	0	1	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	1	1	0	0	0	0
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	1	1	0	0	0	0
32	1	1	0	0	0	0

Reference

1. Habibi, A., "Hybrid Coding of Pictorial Data," IEEE Trans. on Communications, Vol. COM-22, No. 5, pp. 614-624, May 1974.



Fig. 1 512x512 Original Image



Fig. 2 512x512 Coded Image,
2 bits/picture element



Fig. 3 512x512 Coded Image,
1 bit/picture element



Fig. 4 512x512 Coded Image,
1/2 bit/picture element



Fig. 5 256x256 Original Image



Fig. 6 256x256 Coded Image,
2 bits/picture element



Fig. 7 256x256 Coded Image,
1 bit/picture element



Fig. 8 256x256 Coded Image,
1/2 bit/picture element



Fig. 9 512x512 Original Image



Fig. 10 512x512 Coded Image
2 bits/picture element



Fig. 11 512x512 Coded Image
1 bit/picture element



Fig. 12 512x512 Coded Image,
1/2 bit/picture element



Fig. 13 256x256 Original Image



Fig. 14 256x256 Coded Image,
2 bits/picture element



Fig. 15 256x256 Coded Image,
1 bit/picture element



Fig. 16 256x256 Coded Image,
1/2 bit/picture element

DPCM QUANTIZATION ERROR REDUCTION FOR IMAGE CODING*

William K. Pratt
 Michael N. Huhns
 Department of Electrical Engineering
 Image Processing Institute
 University of Southern California
 Los Angeles, California 90007

Abstract

A method for optimally reducing the error incurred during the quantization operation in DPCM image coding is presented. The method is optimum with respect to a mean-square error criterion and is applied a posteriori. The technique is based on knowledge of the multidimensional probability density function of the DPCM difference signal. The quantizer operates on this difference signal to achieve a data compression. The coarser the quantization, the greater the compression, but also the greater the degradation of the encoded image. To minimize the degradation, a reconstruction should utilize all of the knowledge that is available about the difference samples, such as their distribution, the quantization levels, and any correlation which remains after the differencing operation. An estimation equation which embodies this information is derived and solved. The solution is applied to images which have been either DPCM or deltamodulation encoded. The resultant images have lower mean-square error and exhibit an improvement in subjective quality.

Introduction

Differential Pulse Code Modulation (DPCM) is a common coding method for achieving a bandwidth compression for digital image transmission. With this technique, as shown in figure 1, the difference between a pixel value and its estimate is quantized, coded, and transmitted, rather than the pixel value itself. A bandwidth compression, as compared to conventional PCM coding, is possible since the inherent pixel-to-pixel correlation permits a reasonable gross quantization of the DPCM difference signal.

To minimize the degradation resulting from quantization of the DPCM difference signal, the reconstruction unit should utilize all of the available knowledge of the difference signal such as the settings of the quantization levels, the signal distribution, and any residual spatial correlation. This paper describes such a reconstruction procedure which significantly improves the performance of conventional DPCM coders for image coding applications.

DPCM Statistical Model

Experimental evidence (1) indicates that the first order probability density of a DPCM difference signal can be well modelled by a Laplacian density of the form

$$p(x) = [2\sigma]^{-1} \exp\left[-\frac{|x|}{\sigma}\right] \quad (1)$$

where σ is the standard deviation of the difference signal. It has also been found that the DPCM difference signals are correlated with one another. Figure 2a contains a perspective plot of the measured second order probability density for the "girl" image of figure 3. For this image, the average correlation of adjacent difference signals was found to be 0.4. The second order probability density of DPCM difference signals can be modelled by the two-dimensional exponential density function

$$p(x_1, x_2) = \frac{1}{2\sigma_{x_1}\sigma_{x_2}\sqrt{1-r^2}} \exp\left\{-\frac{1}{\sqrt{2(1-r^2)}} \left(\left| \frac{ax_1}{\sigma_{x_1}} - \frac{bx_2}{\sigma_{x_2}} \right| + \left| \frac{ax_2}{\sigma_{x_2}} - \frac{bx_1}{\sigma_{x_1}} \right| \right) \right\} \quad (2)$$

$$r = \frac{E\{x_1 x_2\}}{\sigma_{x_1} \sigma_{x_2}} \quad (3)$$

$$a = \sqrt{1+r} + \sqrt{1-r} \quad (4)$$

$$b = \sqrt{1+r} - \sqrt{1-r} \quad (5)$$

Figure 2b contains a plot of this density function for $r=0.4$ and $\sigma_{x_1}=\sigma_{x_2}$. Higher order probability density models can also be generated to account for the relationships between groups of DPCM difference signals.

DPCM Image Reconstruction

A minimum mean-square error reconstruction of the quantized DPCM difference signals can be obtained by evaluation of the vector conditional expectation

$$E\{\underline{x} | \underline{x} \in R\} = \frac{\int_R \underline{x} p(\underline{x}) d\underline{x}}{\int_R p(\underline{x}) d\underline{x}} \quad (6)$$

where \underline{x} is a vector of DPCM difference signals, R denotes the quantization cell, and $p(\underline{x})$ is the joint probability density of DPCM samples (2). The general solution to eq. (6) is complicated. However, solutions have been obtained for the case in which the two nearest neighbors are utilized in the DPCM image reconstruction. The simplest example is that of Deltamodulation in which a pair of difference signals (x_1, x_2) are each quantized to the interval $(0, \infty)$ or $(-\infty, 0)$, or equivalently, as positive or negative values. The optimum reconstruction values for the pair of DPCM signals is then found to be

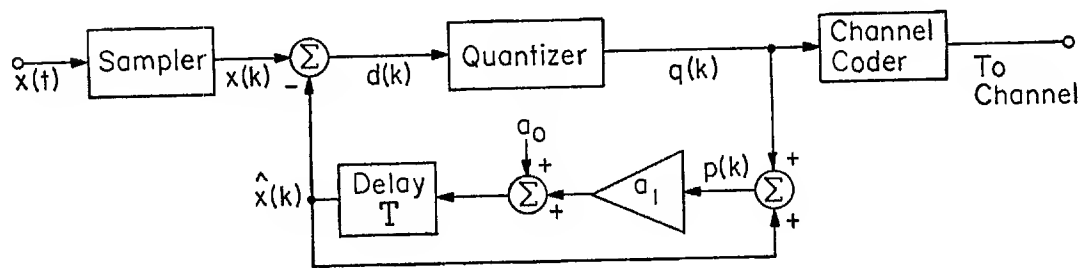
$$\hat{x}_1 = \frac{\sigma_{x_1} [1+2|r|] \operatorname{sgn}(x_1)}{\sqrt{2[2/1+|r| - \sqrt{1+|r|}]}} \quad \text{for } rx_1x_2 > 0 \quad (7a)$$

$$\hat{x}_1 = \sigma_{x_1} \frac{\sqrt{1+|r|}}{2} \operatorname{sgn}(x_1) \quad \text{for } rx_1x_2 < 0 \quad (7b)$$

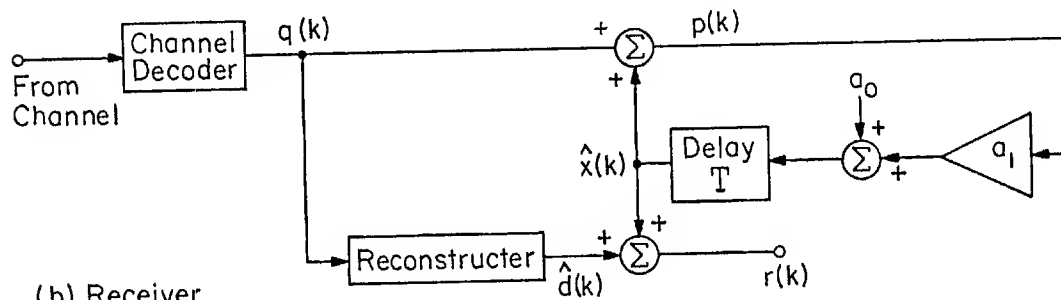
where $\operatorname{sgn}(x_i)$ denotes the sign of x_i .

Simulation Results

A computer simulation has been performed to evaluate the DPCM quantization error reduction process. Figures 3a, 3c, and 3e contain pictures of the girl image DPCM coded with 1, 2, and 3 bits/pixel, respectively, by conventional DPCM coding and reconstruction. In the reconstruction of figure 3b, the present and past DPCM difference signals have been utilized to obtain the reconstruction. This procedure resulted in a 39% decrease in mean square error and a discernible improvement in subjective quality. The reconstructions in figures 3d and 3f utilized the previous DPCM difference signal in the reconstruction. Both image reconstructions provide a lower mean-square error and subjective improvement in quality. However, the relative improvement is smaller because the quantization error is smaller at higher coding rates.

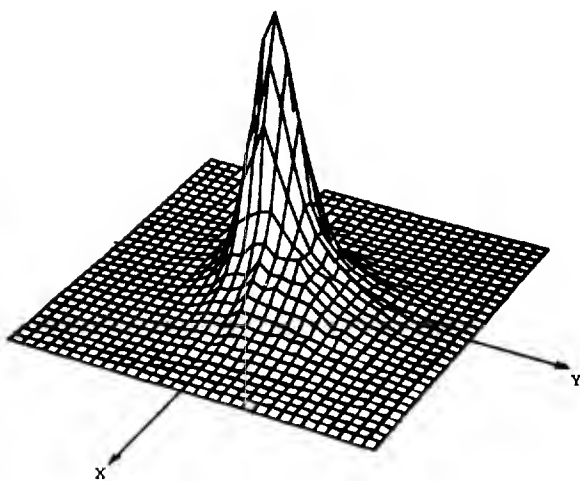


(a) Transmitter

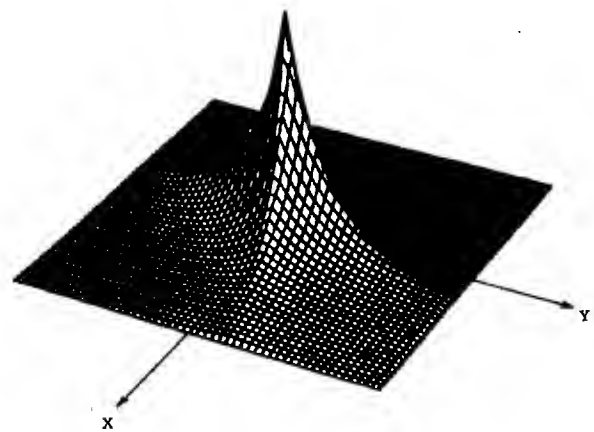


(b) Receiver

Figure 1. Spatial predictive DPCM coding system with quantization restoration



(a) Histogram of the DPCM coded 'girl' image.



(b) Correlated ($r=0.4$) laplacian density function.

Figure 2. Two-dimensional distributions for DPCM simulations.



(a) DPCM 1 bit/pixel



(b) 1 bit/pixel restored



(c) DPCM 2 bits/pixel



(d) 2 bits/pixel restored



(e) DPCM 3 bits/pixel



(f) 3 bits/pixel restored

Figure 3. Minimum mean-square error restoration of DPCM coded images.

Summary

The DPCM quantization error reduction method is a relatively simple processing technique for improving the subjective quality of DPCM coded images and reducing the mean square coding error. The method possesses the advantage that all processing is performed at the receiver, without any modification to the transmitter unit.

References

1. J. B. O'Neal, Jr., "Predictive Quantizing Systems (Differential Pulse Code Modulation) for the Transmission of Television Signals," Bell System Technical Journal, Vol. 45, May/June 1966, pp. 689-721.
2. M. N. Huhns, "Transform Domain Spectrum Interpolation," University of Southern California Image Processing Institute Technical Report, USCIPR Report 530, March, 1974, pp. 28-38.

*This work was supported by the Advanced Research Projects Agency of the Department of Defense and monitored by the Air Force Eastern Test Range under Contract F08606-72-C-0008.

COMBINED SPATIAL AND TEMPORAL CODING
OF DIGITAL IMAGE SEQUENCES

John A. Roese
Naval Undersea Center
San Diego, California 92132

Guner S. Robinson
Image Processing Institute, University of Southern California
Los Angeles, California 90007

Abstract

Interframe coding of television images encompasses techniques which make use of correlations between pixel amplitudes in successive frames. Intraframe coding techniques that exploit spatial correlations can, in principle, be extended to include correlations in the temporal domain.

In this paper, successive frames of digital images are coded using two-dimensional spatial transforms combined with DPCM in the temporal domain. Specific transform techniques investigated are the two-dimensional cosine and Fourier transforms. Due to DPCM encoding in the temporal domain, the hybrid transform/DPCM encoders require storage of only the single previous frame of data.

Hardware implementation of the Fourier transform involves manipulation of complex numbers where the cosine transform does not. However, the Fourier transform is attractive because frame-to-frame motion compensation can be introduced directly in the phase plane by application of appropriate phase correction factors.

Results are presented in terms of coding efficiency, storage requirements, computational complexity, and sensitivity to channel noise.

Introduction

In the design of image coding systems for digital communications channels, the primary objective is to minimize the number of code bits required to reconstruct the image at the receiver. Reduction in the number of code bits transmitted results in reduced channel bandwidth, more rapid transmission of digital images, and lower transmitter power requirements.

Efficient coding of the digital images is accomplished by removal of statistical redundancies that exist within the image. Transform, predictive, and hybrid transform/predictive image coding techniques have been developed to exploit intraframe spatial image redundancies. This paper describes efforts to extend these image coding techniques to coding of time-sequences of digital images transmitted over a digital communications channel. The emphasis has been directed towards definition of an image coding system that exploits temporal as well as spatial image redundancies.

Intraframe Image Coding

The primary techniques that have been developed for intraframe image coding in the spatial domain are transform, linear predictive, and hybrid transform/linear predictive techniques. Operational descriptions of these coding methods are given below.

Transform Image Coding

The basic premise of transform image coding is that the transform domain representation of an image has an energy distribution that is more compact and therefore easier to efficiently code than the spatial domain version. In transform coding systems, a one- or two-dimensional linear transform of an image line segment or block is performed at the coder. The transform coefficient statistics are computed prior to quantization and coding for transmission. After decoding at the receiver, an inverse transform is taken to obtain a reconstructed image. Transforms that have proven useful for this application include Fourier, cosine, Hadamard, Slant, and Karhunen-Loeve.[1,2,3] It should be noted that the two transforms can be different for the two spatial directions.

Two-dimensional transforms have the inherent disadvantage of requiring an intermediate memory to store the transform coefficients computed in one direction while the transform is being computed in the other direction. Transform coding techniques have been explored extensively both theoretically and by simulation. It has been shown that significant bit rate reductions can be achieved in many applications with minimal image degradation.

Simulation results indicate that a bit rate reduction to 1.5 bits/pixel can be obtained for monochrome image transform coding in 16×16 picture element (pixel) blocks, while color images require about 2.0 bits/pixel.[4] The bit rate can be reduced further by making the transform coding system adaptive.

Predictive Image Coding

The high degree of correlation between a given pixel value and its nearest neighbors allows an image to be efficiently represented by coding only the difference between each pixel value and its predicted value. The predicted pixel value is based on previously scanned pixel values. In a differential pulse code modulation (DPCM) system, the predicted value of each pixel is subtracted from its actual value and this difference is quantized and transmitted. Quantization of the data is done with a quantizer designed for the probability density of the difference signal. Thus, coding by use of a DPCM system requires a knowledge of the data statistics. The basic operation of the DPCM coder is to generate an uncorrelated signal which is then encoded by a memoryless quantizer for transmission. At the receiver, the quantized difference signal is combined with its predicted value to form the reconstructed pixel value. Basic DPCM image coding systems provide good quality at about 3 bits/pixel. Adaptive systems, in which parameters of the quantizer and predictor adapt to the image content, require about 2 bits/pixel.[5]

Hybrid Image Coding

Analysis of transform and predictive coding techniques has shown that both techniques possess attractive characteristics and certain limitations. Transform coding techniques achieve good performance at low bit rates, show less sensitivity to data statistics (picture-to-picture variations) and are less vulnerable to channel noise. Predictive coding systems are superior to transform techniques in terms of equipment complexity, memory requirements, and performance at high bit rates. Some limitations of predictive systems are their sensitivity to data statistics and to channel error.

An intraframe hybrid coding system that combines the attractive features of both transform and predictive coding systems has been devised.[6] In this system, a one-dimensional transform is followed by DPCM linear predictive coding of the transform domain coefficients. At the receiver, the transform coefficients are decoded and a replica of the original image is reconstructed by an inverse transform.

Hybrid Interframe Image Coding

Interframe coding of digital image sequences encompasses those techniques which make use of the high correlation that exists between pixel amplitudes in successive frames. Intraframe coding techniques that exploit spatial correlations can, in principle, be extended to include correlations in the temporal domain. Previous research in the area of three-dimensional Fourier and Hadamard transformations has indicated that bit rates can be reduced by a factor of five by incorporating correlations in the temporal direction.[7] However, three-dimensional transform systems are unattractive as they use large amounts of data storage and require excessive computations.

To alleviate the problems associated with three-dimensional transform systems, new hybrid (two-dimensional transform)/DPCM image coding systems have been developed.[8] These systems utilize both spatial and temporal correlations while greatly reducing memory storage and computational requirements. The block diagram for a hybrid (two-dimensional transform)/DPCM system is shown as figure 1. In present implementations of this system, either a two-dimensional cosine or Fourier transformation is performed on 16×16

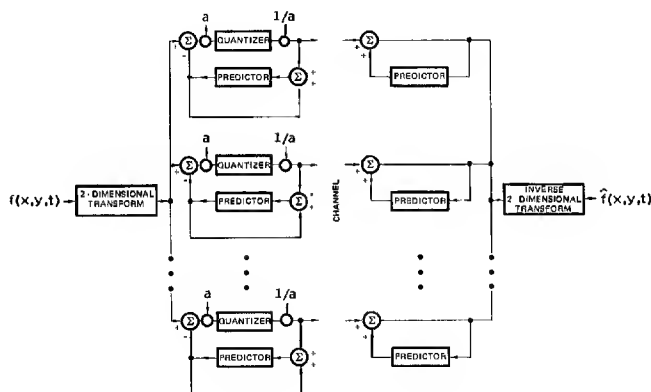


Fig. 1. Hybrid (two-dimensional transform)/DPCM coder.

subblocks. DPCM linear predictive coding in the temporal domain is then applied to the transform coefficients of each subblock. For notational convenience, the hybrid interframe coders employing two-dimensional Fourier transforms will be denoted as FFD and those using two-dimensional cosine transforms as CCD. The FFD and CCD coders are adaptive in the sense that statistics of the transform coefficient differences of each subblock are computed prior to encoding the transform coefficients in the temporal direction by parallel DPCM coders. At the receiver, the transmitted transform coefficients are decoded and a replica of each frame is reconstructed by the appropriate inverse two-dimensional transformation. These systems require only a single frame of storage and involve significantly less memory and fewer computations than three-dimensional transform coding techniques.

Operational Modes

At least three operational modes have been identified for the hybrid interframe coding systems. These operational modes depend on the initial conditions assumed for the predictive coder. The initial conditions are:

- a. No apriori information available at the receiver,
- b. Limited information (such as mean, variance and temporal correlations based on a statistical model) available at the receiver, and
- c. First frame available at the receiver.

In the no apriori information available case, several frames are required for the hybrid coder to settle. However, it has been experimentally verified that in the remaining two cases, nearly stable coder performance is achieved within the first 4 to 6 frames. From operational considerations, the third set of initial conditions is the most realistic as periodic full frame updating will be required to eliminate the cumulative effects due to channel noise.

Mathematical Formulation

Let $f(x,y)$ denote a two-dimensional array of intensity values on an $N \times N$ subblock of a digital television image of size $M \times M$. Typical values for M and N are 256 and 16, respectively. Let $F(u,v)$ be the two-dimensional array obtained by taking the two-dimensional transform of $f(x,y)$. In the case of the two-dimensional discrete Fourier transform, the expressions relating $f(x,y)$ and $F(u,v)$ are

$$F(u,v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \exp \left[-\frac{2\pi i}{N} (ux + vy) \right] \quad (1)$$

$$u,v = 0, 1, \dots, N-1$$

and

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) \exp \left[+\frac{2\pi i}{N} (ux + vy) \right] \quad (2)$$

$$x,y = 0, 1, \dots, N-1$$

where N is the size of the square subblock, $f(x,y)$ is the image intensity function in the spatial domain, x and y are spatial coordinates, $F(u,v)$ is the Fourier transform, and u and v are spatial frequencies.

For image processing applications, $f(x,y)$ is a positive real function representing brightness of the spatial sample. The two-dimensional Fourier transform of a real-valued function has the following conjugate symmetry property:

$$F^*(u,v) = F(N-u, N-v) \quad , \quad u,v = 1, 2, \dots, \frac{N}{2} - 1$$

The Fourier transform consists of $2N^2$ components, i.e., the real and imaginary or magnitude and phase components of each spatial frequency. However, as a result of the conjugate symmetry properties mentioned above, only N^2 components are required to completely define the Fourier transform.[9]

In the case of the Fourier transform, a shift in the spatial-domain variables results in a multiplication of the Fourier transform of the un-shifted image by a phase factor. If the input image $f(x,y,t_1)$ is shifted by the amount x_0 in the x -direction and y_0 in the

y-direction between times t_1 and t_2 , then the Fourier transform of the shifted image is given by

$$F(u, v, t_2) = F(u, v, t_1) \exp \left[\frac{2\pi i}{N} (ux_0 + vy_0) \right] \quad (3)$$

This shifting property is expected to be useful in detecting and compensating for effects of motion between frames since many types of motions such as panned motion produce significant changes in phase components and small changes in amplitude components. Thus, compensation for platform motion may be implemented directly in the array of phase components by application of appropriate phase correction factors.

The two-dimensional Fourier transform $F(u, v)$ of a spatial signal function $f(x, y)$ is separable, i.e., it can be computed as two sequential one-dimensional transforms since the Fourier kernel,

$$\exp \left[\pm \frac{2\pi i}{N} (ux + vy) \right]$$

is separable and symmetric. Thus, the basic one-dimensional discrete Fourier transform that must be performed is

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp \left(- \frac{2\pi i}{N} ux \right) \quad (4)$$

for $u=0, 1, \dots, N-1$.

In the case of the discrete Cosine transform[3], the one-dimensional transform is

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos \frac{(2x+1)u\pi}{2N} \quad (5)$$

for $u=0, 1, \dots, N-1$. The form of eq. (5) differs from that of reference [3] only by a normalization constant. The cosine transform is also separable and a two-dimensional discrete cosine transform of an $N \times N$ subblock results in N^2 real coefficients.

Quantization

Experimental evidence derived from transmission of a typical "head and shoulders" picture telephone scene has shown that the frame difference signal has a probability density closely approximated by a double sided exponential function.[10] The optimum minimum mean square error quantizer for this distribution has been found to be a uniform quantizer combined with a companding of the frame difference signal.[11]

Since the variances of the transform domain coefficient differences are different, it is necessary to use different quantizer parameters for each one. Each coefficient difference signal is allocated a number of bits proportional to its estimated variance in accordance with an optimum bit assignment algorithm.

Fidelity Criteria

In figure 1, differences between input signal $f(x, y, t)$ and output signal $\hat{f}(x, y, t)$ are due to two sources: quantization errors and errors due to channel noise. To evaluate coding efficiency of the hybrid encoders, two objective criteria were used. The first criterion, NMSE, is a measure of the mean square error between $f(x, y, t)$ and $\hat{f}(x, y, t)$ averaged over an entire frame of size $M \times M$. Normalization is achieved by dividing the mean square error by the mean signal energy within the frame.

$$NMSE = \frac{\frac{1}{M^2} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |f(x, y, t) - \hat{f}(x, y, t)|^2}{\frac{1}{M^2} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |f(x, y, t)|^2} \quad (6)$$

The second criterion, SNR, measures the ratio of peak-to-peak signal to RMS noise:

$$SNR = -10 \log_{10} \frac{\frac{1}{M^2} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |f(x,y,t) - \hat{f}(x,y,t)|^2}{255^2} \quad (7)$$

Figures 2 and 3 are graphs illustrating the coding efficiency of the hybrid FFD and CCD coders at various bit rates in the interval 0.1 to 1.0 bits/pixel/frame. To perform this series of experiments, a 256 x 256 resolution data base consisting of 16 consecutive frames of a 24 frames per second (fps) motion picture was digitized. Initial conditions assumed were that the first frame was available at the receiver.

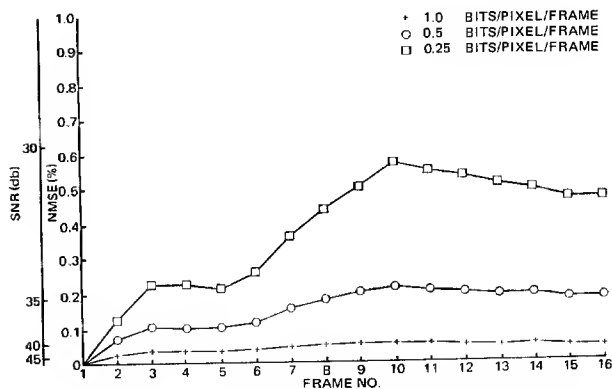


Fig. 2. Fourier/Fourier/DPCM coder at various bit rates.

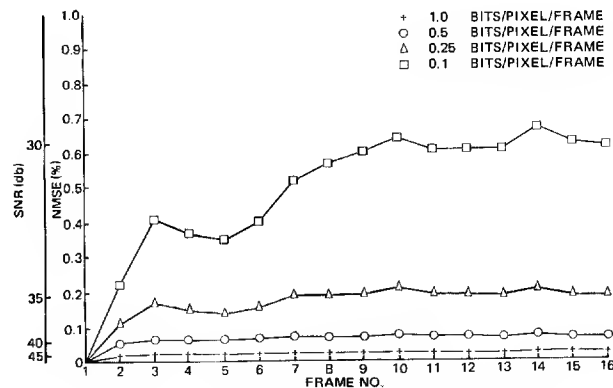


Fig. 3. Cosine/cosine/DPCM coder at various bit rates.

Photographs of frame number 16 after coding by the FFD and CCD coders at average pixel bit rates of 1.0, 0.5, 0.25, and 0.1 are shown as figures 4 and 5. The results shown in figure 4 for the FFD coder were obtained by coding the real and imaginary components of the Fourier coefficients by assigning half of the available bits to each component.



1.0 bits/pixel/frame 0.5 bits/pixel/frame 0.25 bits/pixel/frame 0.1 bits/pixel/frame

Fig. 4. FFD coder for frame 16.



1.0 bits/pixel/frame 0.5 bits/pixel/frame 0.25 bits/pixel/frame 0.1 bits/pixel/frame

Fig. 5. CCD coder for frame 16.

Noise Immunity

Performance of the FFD and CCD hybrid interframe coders was investigated in the presence of channel noise. In order to study the effect of channel noise, a binary symmetric channel was simulated. The channel is assumed to operate on each binary digit independently, changing each digit from 0 to 1 or from 1 to 0 with probability P_{ce} and leaving the digit unchanged with probability $1-P_{ce}$. At the receiver, the encoded picture is reconstructed from the string of binary digits, including errors, transmitted across the channel.

Degradations due to channel noise probabilities, P_{ce} , of zero, 10^{-3} and 10^{-2} for the FFD and CCD coders at average bit rates of 1.0 and 0.25 bits/pixel/frame are shown in figures 6 through 9. The generally monotonically increasing character of these curves illustrates the fact that once an error has occurred, it tends to propagate in the temporal direction until corrected by a frame refresh.

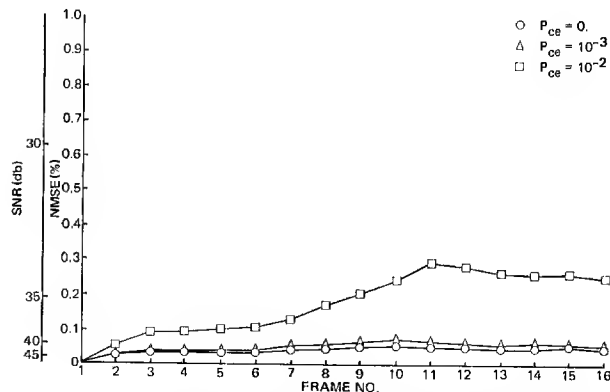


Fig. 6. Effects of channel noise for Fourier/Fourier/DPCM coder at an average 1.0 bits/pixel/frame.

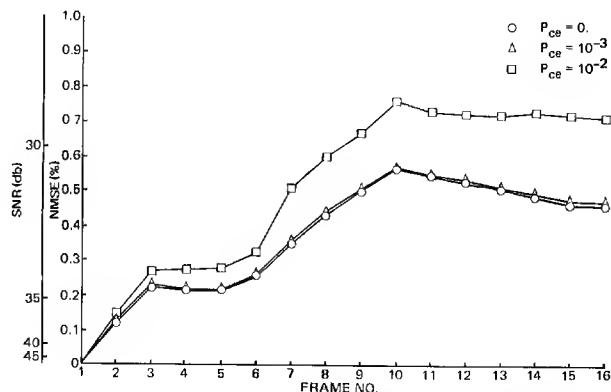


Fig. 7. Effects of channel noise for Fourier/Fourier/DPCM coder at an average 0.25 bits/pixel/frame.

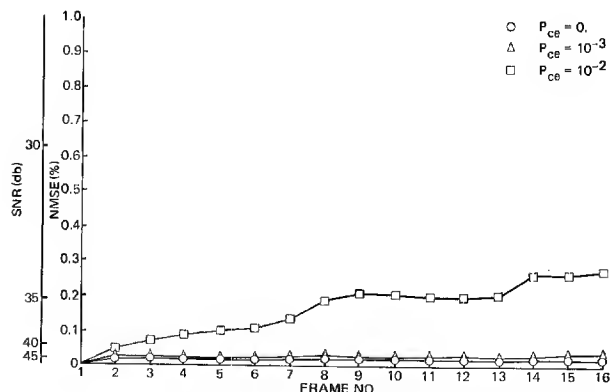


Fig. 8. Effects of channel noise for cosine/cosine/DPCM coder at an average 1.0 bits/pixel/frame.

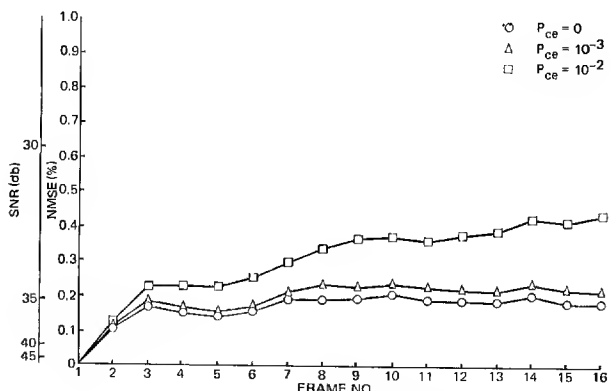


Fig. 9. Effects of channel noise for cosine/cosine/DPCM coder at an average 0.25 bits/pixel/frame.

Photographs corresponding to average bit rates of 1.0 and 0.25 bits/pixel/frame for the FFD and CCD coders with channel error probabilities of 10^{-3} and 10^{-2} are shown in figures 10 and 11.

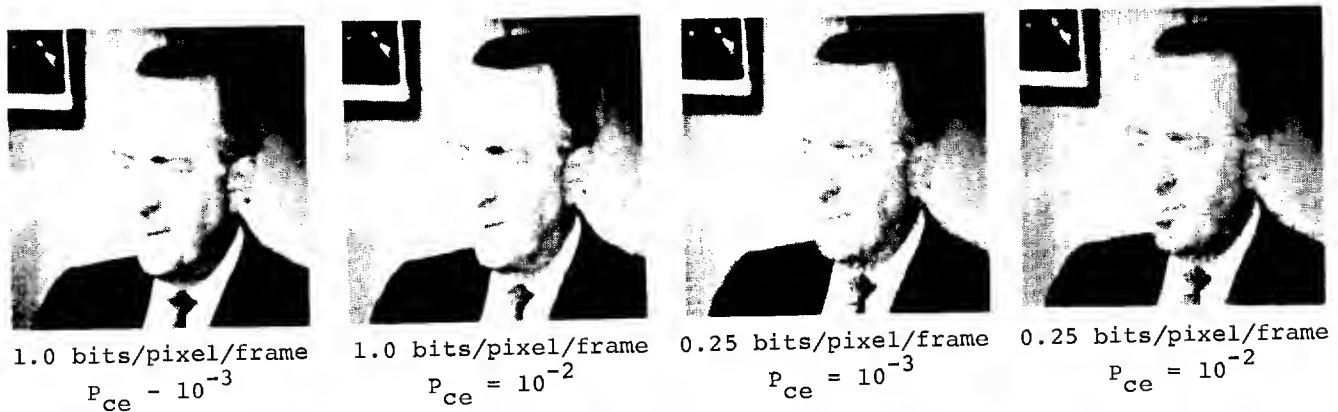


Fig. 10. FFD coder with channel noise.

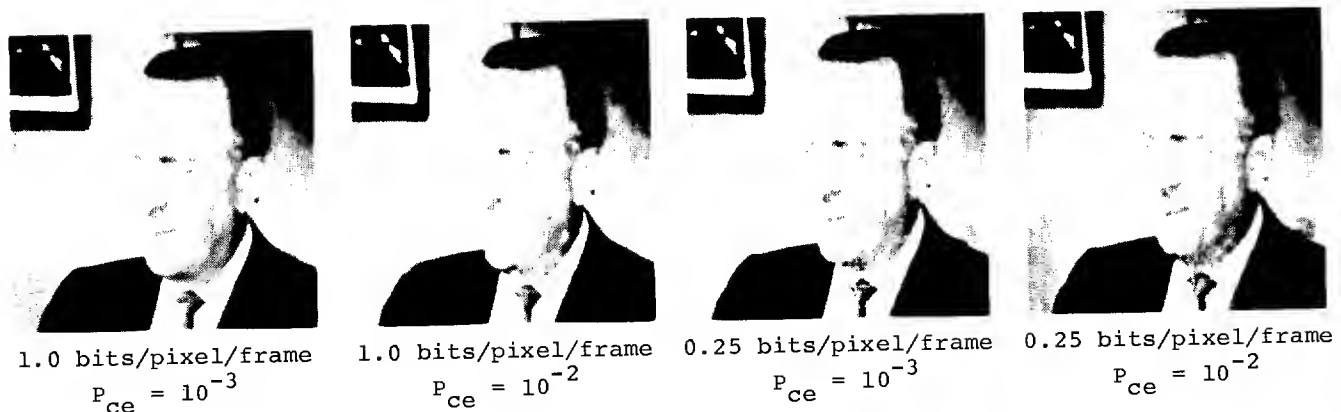


Fig. 11. CCD coder with channel noise.

Bit Transfer Rate

In keeping with the previously mentioned objective of minimizing the number of bits transmitted while retaining image fidelity, a series of experiments was performed in which certain bit transfer rates (BTR) across the channel were fixed. The BTR is defined as the product of average pixel bit rate per frame and frame rate and has units of bits/pixel/sec.

$$BTR = (\text{bits/pixel/frame}) \times (\text{frames/sec}) \quad (8)$$

The available 16 frame test data base was extracted from a 24 fps motion picture sequence. By employing frame skipping techniques, temporal subsampling was used to simulate short 12, 8 and 6 fps sequences from the 16 frame test data base.

Average bit rates in the interval 0.083 to 1.333 bits/pixel/frame were used in conjunction with the four frame rates mentioned above to perform simulations with BTR values of 8, 6, 4 and 2 bits/pixel/sec. The results of these experiments are shown as figures 12 through 15, respectively. For all cases examined, the graphs show that reduced frame rates produce smaller NMSE values for the individual frames coded. This indicates that reductions experienced in frame-to-frame correlations due to temporal subsampling are completely compensated for by the increased number of bits available for coding. However, subjectively, reduced frame rates tend to result in jerky subject motion. This is most apparent for rapidly moving objects in the field of view and is of lesser consequence for slowly changing scenes.

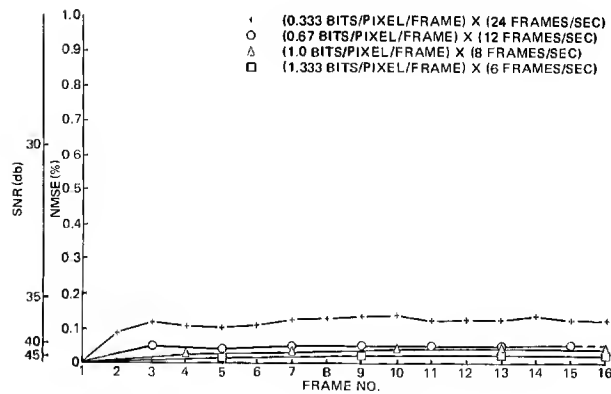


Fig. 12. Cosine/cosine/DPCM coder at bit transfer rate of 8 bits/pixel/sec.

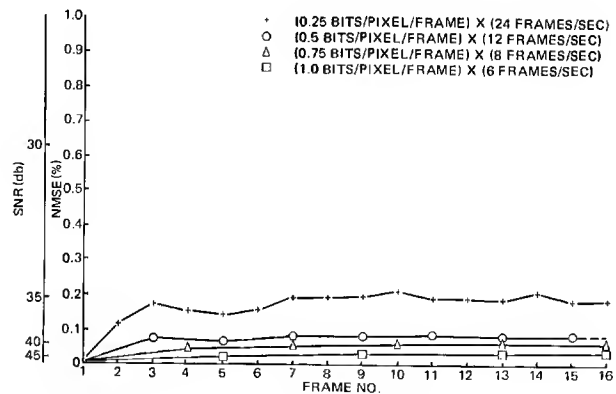


Fig. 13. Cosine/cosine/DPCM coder at bit transfer rate of 6 bits/pixel/sec.

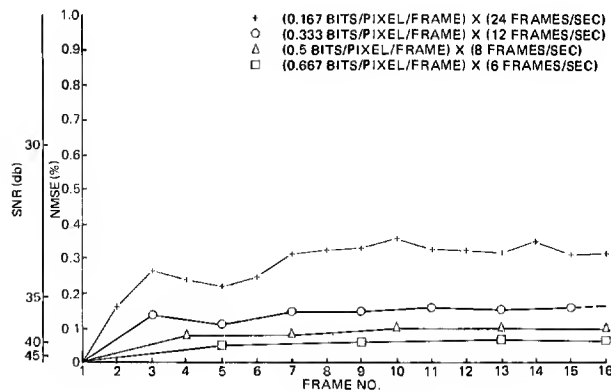


Fig. 14. Cosine/cosine/DPCM coder at bit transfer rate of 4 bits/pixel/sec.

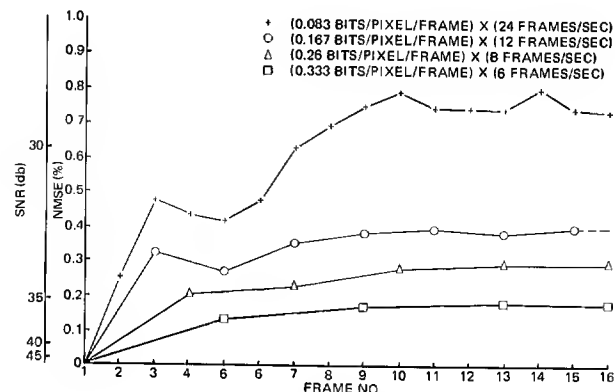


Fig. 15. Cosine/cosine/DPCM coder at bit transfer rate of 2 bits/pixel/sec.

Conclusions

Based on theoretical and experimental results obtained to date, two main conclusions have been reached. The first is that exploitation of temporal correlations in addition to spatial correlations has been demonstrated to be a viable technique for coding sequences of digital images. This fact is demonstrated by a comparison of the average bit rates required for the interframe cosine/cosine/DPCM and the existing intraframe cosine/DPCM coders to achieve the same level of NMSE performance. The sixteenth frame of the test data base was chosen for comparison and was coded at an average 0.25 bits/pixel by the interframe cosine/cosine/DPCM coder. When using the intraframe cosine/DPCM coder, it was necessary to code this frame at a bit rate of more than 2 bits/pixel to achieve the same NMSE.

The second conclusion is that the performance of the hybrid interframe coders investigated are heavily data dependent. In the case of the 16 frame head and shoulders test data base, good coding performance was achieved since subject movement was restricted to a relatively small portion of the image. However, coding performance with a different aerial data base was degraded from the previous case due principally to platform motion which caused frame-to-frame pixel amplitude variations across the entire image. Since the performance of the hybrid interframe coders is dependent on temporal correlation, a reduced level of performance is to be anticipated for image sequences distorted by motion.

References

1. P. A. Wintz, "Transform Picture Coding," Proceedings IEEE, Vol. 60, No. 7, pp. 809-820, July 1972.
2. W. K. Pratt, L. R. Welch and W. Chen, "Slant Transforms for Image Coding," IEEE Transactions on Communications, Vol. COM-22, No. 8, pp. 1075-1093, August, 1974.
3. N. Ahmed, T. Natarajan, K. R. Rao, "Discrete Cosine Transform," IEEE Trans. on Computers, Vol. C-23, No. 1, pp. 90-93, January 1974.
4. W. K. Pratt, "Spatial Transform Coding of Color Images," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, pp. 980-992, December, 1971.
5. A. Habibi, "Comparison of nth -order DPCM Encoder with Linear Transformation and Block Quantization Techniques," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, Part I, pp. 948-956, December 1971.
6. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transactions on Communications, Vol. COM-22, No. 55, pp. 614-624, May 1974.
7. A. G. Tescher, "The Role of Phase in Adaptive Image Coding," Ph.D. Thesis, University of Southern California, Electrical Engineering Department, January 1974. Published as Report 510, University of Southern California, Image Processing Institute.
8. J. A. Roese, W. K. Pratt, G. S. Robinson and A. Habibi, "Interframe Transform Coding and Predictive Coding Methods," Proceedings of 1975 International Conference on Communications (ICC 75), Vol. II, Paper 23, pp. 17-21, June 16-18, 1975.
9. G. S. Robinson, "Orthogonal Transform Feasibility Study," NASA Final Report NASA-CR-115314, N72-13143 (176 pp.) (submitted by COMSAT Labs. to NASA Manned Spacecraft Center, Houston, Texas) November 1971.
10. A. J. Seyler, "Probability distributions of television frame differences," Proc. IREE, Australia, pp. 355-366, November 1965.
11. B. Smith, "Instantaneous companding of quantized signals," Bell System Technical Journal, Vol. 36, pp. 653-709, May 1957.

Acknowledgement

This research was supported by the Advanced Research Projects Agency of the Department of Defense and monitored by the Air Force Eastern Test Range under Contract R08606-72-0008 and by the Naval Undersea Center, San Diego, California, under Contract N00123-75-C-1192.

Session V
RELATED TOPICS

Chairman
HARRY C. ANDREWS
Image Processing Institute, University of Southern California

LASER SCANNER/RECORDERS
FOR IMAGE TRANSMISSION AND COMPUTER PROCESSING

W.F. Schreiber, U.F. Gronemann* and D.E. Troxel
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Abstract

Image processing by computer or otherwise requires cost-effective means of scanning and recording high quality images and of interfacing the imaging devices with computers and communication channels. Based on our previous work in newspaper facsimile and computer editing of news photos, we have developed a large-format high-resolution system, interfaced to a PDP-11 processor as well as the Bell DDS Communications channel which operates at 56 kB/s. One configuration, suitable for medical X-rays, features variable resolution up to 250 pels/inch over 14 inches and 1000 pels/inch over 4 inches and a 256 step density scale. Dry silver film with on-line development produces ready-to-use output of high diagnostic usefulness.

The technology used includes low-power internally modulated HeNe lasers, pre-objective scanning with a galvanometer-driven mirror, a flat-field paper/film drive, and silicon photodetectors.

This work was supported by NIGMS Grant 5 P01-GM19428-03S1 and by USPHS Grant GM018674 (through the Departments of Radiology of Peter Bent Brigham Hospital and Beth Israel Hospital).

Introduction

For the past twenty years, the image processing laboratory in what is now the Cognitive Information Processing Group of the Research Laboratory of Electronics has been engaged in research related to image coding, transmission, and processing. In the course of that work we have developed a wide variety of scanners, including rotating drum, cathode ray tube, vidicon, and most recently, laser systems. We have been attracted to the latter because higher performance may be obtained at lower cost.

In 1970, we began the development of a new newspaper picture distribution system for the Associated Press. The transmitters and receivers, which resulted are now in production by Harris Electronics under the trade name Laserphoto⁽¹⁾, and will, by sometime in 1976, replace all the mechanical systems now in use by the Associated Press. Comparable in cost to the older systems, they offer advantages in reliability and, above all, in picture quality. They scan at about 100 lines/inch, 100 lines/minute in an 11" wide format, using 3M Dry Silver Paper and an integral heat processor.

Subsequently, we have developed a variable format higher performance system for medical radiographs⁽²⁾, having a resolution of 250 lines/inch along a scanned line up to 14" wide, with output on 3M Dry Silver film. This system employs a 56 kilobit/sec digital transmission channel between the transmitter and receiver modules.

Both of these systems obtain much of their economy from a flat-field scanner, which permits use of a flat-bed paper/film drive. The flat field is achieved by pre-objective scanning - i.e., deflecting a collimated laser beam by a rotating or oscillating flat mirror placed at the entrance aperture of a focusing lens.

The system described in this paper is a further development of the previous work, differing in that we are employing a specially designed objective for higher resolution and in addition are providing interfaces both to a PDP-11 computer system and the Bell DDS digital communication channel. In this way we hope to explore the use of the scanner for a generalized computer processing of very high quality large format pictures as well as to investigate the potential applications for rapid transmission of medical X-rays of essentially unimpaired diagnostic utility.

Optical System

Scanning Optics

Some considerations relative to pre- and post-objective scanning are discussed in Beiser's excellent paper⁽³⁾. The principles may easily be understood by reference to Figure 1 which shows a typical laser optical system which projects a focussed spot on the page at P.

Assuming diffraction-limited performance, the spot size at P depends only on the f/ number, F/d, and the wavelength, together with a small factor which depends on the energy distribution in the laser beam. Placing a deflection member at B causes the objective lens to be used on-axis only, which is highly desirable, but produces a curved focal surface, as shown in Figure 2. A deflecting member at A requires that the objective function

* Currently - Department of Radiology, Harvard Medical School, Boston, MA.

over some scan angle, which is, of course, more difficult, but produces a flat focal surface, as shown in Figure 3, if the lens has a flat field. Note that in both cases the objective is focussed at infinity.

The essential question is how the difficulties of the flat-field lens design trade off against the advantages of a flat-bed paper/film transport. As it happens, camera lenses must also have a flat field, so that optical designers have been working on this problem for well over one hundred years. Thus low-cost standard lenses for this application exist down to about $f/32$, which was used in the first version of the X-ray system. The Laserphoto requirement can be met at about $f/100$. On the other hand, a very high performance, very expensive system of this type has been built at $f/10$ ⁽⁴⁾. For our new scanner, we are using a specially-designed, but still reasonably inexpensive $f/18.5$ objective.

The initial Laserphoto scanner (but not Harris's production design), our first-version X-ray system, and the above-noted $f/10$ system are all of the double-pass type, as shown in Figure 4. The double pass configuration permits the use of a smaller diameter objective for a given f /number and scan angle, and is very useful especially when utilizing existing lens designs. The particular system used in the Laserphoto apparatus, in which the first pass beam is incident on the scanning mirror outside the plane of the second pass beam, and in which the scanned line in the copy is corrected for straightness by rotating the mirror about an axis inclined to its surface, is the subject of U.S. Patent 3,868,167.

Our new $f/18.5$ system uses a single pass arrangement, shown in Figure 5 together with the light collecting optics which are discussed below.

Light Collecting Optics

In order to achieve the highest possible signal/noise ratio in the video signal, the largest possible proportion of the laser light incident on the film must be collected on the smallest possible detector. In small format systems, a condensing (positive) lens is placed behind the film in order to image the exit aperture of the objective lens onto a photodetector. This is not practical for a 14" wide field, so we originally allowed the transmitted light to fall on a diffuse white reflector as in Figure 6. A strip of solar cells placed near the white surface can collect up to about 10% of the light. [To place the cells directly in the path of the scanning beam makes the video amplitude dependent on the point-to-point sensitivity of the cells and would require very careful correction to avoid stripes in the picture. The use of single crystal high resistivity P-N or P-I-N diodes is possible, but very expensive.]

Since the noise depends very strongly on the area of the cells, it would be better to try to collect the light on a smaller cell. We are now attempting to do this with the large cylindrical collector shown in Figure 5 which images the exit aperture of the focusing lens onto a small photodetector. This appears to be satisfactory, although there is still some question about the cost of a reflector of adequate quality.

Spot Size Control

For any particular resolution (namely, spatial pel density) there is an optimal size and shape of the scanning spot.

Since the electrooptical signal is proportional to the convolution of the spot spatial power distribution and the picture transmission function, the finite size of the spot serves as a low-pass spatial filter. The sampling theorem indeed requires that spatial frequencies be effectively limited (in any direction) to below half the sample frequency. Theoretically, if the original picture contains higher frequencies, perfect filtering would be achieved by a spot whose distribution in x and y is $(d/\pi x) \sin(\pi x/d)$, where d is the spacing between pel positions. Practically the spot distribution is nearly gaussian, i.e. of the form $\exp(-r^2/a^2)$, and the only variable parameter is the nominal radius, a . If we choose $a = d/2$, the ideal distribution is approximated; however some attenuation occurs also in the desired pass-band. On the other hand, violating the sampling theorem by a small factor does not noticeably distort any real picture; only high-frequency periodic structures (as in test patterns) become "aliased", and very sharp edges "move" a small distance. Thus, in order to maximally preserve small low-contrast detail, the spot is made somewhat smaller than indicated above. The value chosen is $a = d/2\sqrt{2}$. The pel spacing is then equal to the spot diameter as measured between the $1/e_2$ points. For example, for 1000 pels/inch resolution, a is about 8.5 nm.

There is one additional consideration: Since the video signal is low-pass filtered (to minimize noise), and since this filter also does not have an ideal flat characteristic with sharp-cutoff, it is desirable to reduce or eliminate the optical filtering in the horizontal direction, so as to minimize attenuation of useful high frequencies. Ideally the spot should have the shape of a vertical line, i.e. be horizontally much sharper than indicated. Practical design considerations yield an elliptical gaussian spot, with " a " in the vertical direction being twice that in the horizontal direction.

The spot size is determined by the focal length, f , of the primary lens (no. 1 in Figure 1), according to the formula $a = \lambda f/2\pi b$, where b is the nominal radius of the laser beam entering the lens. This f may be either positive or negative; in the design incorporating the small objective lens (with double pass), it was convenient to use a positive primary lens, while in the design with the large special objective lens, a negative primary lens is used. The spot ellipticity is achieved by making the entering beam elliptical,

b being vertically half the horizontal value. This is done by means of a pair of cylindrical lenses in a beam-contraction configuration.

The actual selection of spot size is done by inserting the appropriate primary lens (conveniently mounted) into the optical path in the assigned place. Where the aperture of the objective lens is a limiting factor, the spot for the 1000 pel/inch resolution is made round by removing the cylindrical lenses.

Electronic System

Circuitry

Operation of the electronic circuitry is best understood by reference to the block diagram, Figures 7 and 8.

In the transmitter an internally modulated HeNe laser is switched on and off (chopped) at 56 kHz. A feedback system monitors the first harmonic of laser beam intensity and compares it to a reference. The laser modulator is driven by the error signal.

The photodiode on which the information-bearing transmitted (or reflected) light beam impinges is connected, in parallel with an inductor, to the input of a current-to-voltage converting preamplifier. The photodiode is biased to a voltage such as to give a minimum combination of noises due to dark current and due to capacitance (affecting thermal noise in preamp). The inductor "tunes out" the photodiode capacitance at the chopping frequency, to maximize signal amplitude. The preamplifier has a transconductance-type low-noise input stage. Its output is further amplified by a band-pass amplifier then demodulated by a synchronous demodulator and the output of the latter applied to a low-pass filter amplifier and thence to a logarithmic amplifier. The log amplifier thus produces a voltage proportional to the density of each point being scanned. This voltage is sampled, at a rate that is a submultiple of the 56kHz clock, and converted to a binary code by means of a moderately fast analog-to-digital converter.

Other circuits include a sweep generator, a galvanometer-motor driver, a stepping-motor driver and miscellaneous digital control and interface circuits. The sweep generator produces a sawtooth signal, by means of a bootstrap-type integrator, with electronically switchable slope values. This signal is applied to the galvanometer driver to provide the horizontal beam deflection. The beginning and end of each sweep is timed by the control circuit. The latter also provides pulses to trigger the stepping-motor driver, and thus to advance the film past the scanning beam. The control circuit, as well as the digital output circuit, will be discussed in the following paragraphs.

The receiver circuitry is essentially identical to that of the transmitter except, of course, for the signal path. Binary data as well as clock pulses, are received from the communication channel (or computer interface) and assembled into bytes. These are converted into analog voltages by the digital-analog converter and applied to a non-linear amplifier. The NLA reconverts the logarithmic signal scale to that of light intensity and compensates for the D-logE curve of the film in such a way that the output film density would be equal to the input film density. The "marking power" knob controls the final signal amplitude and essentially calibrates the system for the film sensitivity.

An internally modulated laser, similar to the one in the transmitter, is used to produce the modulated light beam to expose the film. In this case, the error signal which drives the modulator is the difference between the laser beam intensity, as measured by the light monitor, and the signal intensity as obtained from the "marking power" control.

The temperature control of the oven maintains a stable processing temperature for the dry silver film, to achieve a suitable film characteristic. The oven temperature is adjustable to compensate for varying film transport speeds which may result from changing the scan parameters.

Digital Interface

The system has been designed to operate in two main transmission modes: (1) serial-by-bit - primarily for facsimile transmission, and (2) serial-by byte - primarily for computer input and output.

In the first mode, data are transmitted and received over a single channel as a continuous stream of bits, synchronously with clock pulses. The latter are either generated in the transmitter and transmitted over a subchannel or, in the case of a synchronous digital communication system, are obtained from that system. The current version of the equipment transmits at the internal clock rate of 56 kHz (which is that of Bell System's DDS wide-band channel); data are transmitted as TTL levels over one wire and clock pulses (also TTL) over another wire. This will be easily converted to interface with a DDS channel when that becomes practical.

In the second mode, 8-bit bytes (each representing one picture element) are transmitted to, or received from the computer I/O interface; input byte timing is under transmitter control, whereas output timing is controlled by the interface module. Present design is for interfacing a PDP-11 through a DMA channel (which includes a single full-word buffer). At the end of each scan line a regular interrupt occurs. This enables the computer to re-allocate buffer memory and possibly a new output (or input) device channel. If the computer or peripheral device can not keep pace, the film advance to a new line can be delayed by one line-scan period.

In either mode, the data consist essentially of a stream of bytes, representing negative density values. During retrace time (between lines) the byte value is set at

maximum (corresponding to white - to cause blanking in the receiver). At the beginning of a picture one particular code byte is transmitted - to denote (in the case of mode 1) the start of transmission. This is followed by a certain number of bytes that carry (in an error-correcting code) the scan parameters, i.e. raster width and height (in terms of number of picture elements), and transmission parameter, i.e. number of bits per byte of data (in case of mode 1). Then the scanning begins and the data bytes are transmitted, as described above. Transmission stops when the preset number of lines has been scanned.

Scan Parameters and Control

The following variable parameters are controllable from the front panel of the transmitter unit:

- (a) Resolution. Five values are available: 1000, 500, 250, 125, and 31 elements per inch. These refer to both horizontal and vertical resolution. The effect is that horizontally the sweep rate is selected to be inversely proportional to the resolution, while vertically the film is advanced for each line a number of steps inversely proportional to the resolution. Normally one would also switch the optics so as to select the appropriate spot size, namely 1X1, 1X2, 2X4, 4X8 or 16X32 mils, respectively. Since the length of a scan line is limited to 4096 elements, the scan width at maximum resolution is limited to about 4 inches, and at the next resolution to about 8 inches.
- (b) Sampling Rate: Three values are available: 7000, 14,000, and 28,000 samples (i.e. bytes or elements or pels) per second. In effect, the scanning speed is made proportional to these values. In the case of mode 1 transmission, where the bit rate is constant, the number of bits per byte is variable, namely 8, 4, or 2. In the case of mode 2 transmission, the rate is limited by the computer system handling capacity and by the desirable density level and the acceptable noise level. For the lowest resolution (31 pels/in) only 7000 pels/sec is available owing to mechanical limitations on scanning velocities. Note: In the first version of the instrument eight sampling rates were available corresponding to any number (between 1 and 8) of bits/byte.
- (c) Scan Format: Height and width are settable, independently, at any multiple of 128 pels up to 4096 pels width (and not over 14 inches, as limited by the optics) and 8192 pels height.

The following controls (d,e,f) do not affect transmission and reception, but enable the operator to direct the transmitter to scan a desired section of a film.

- (d) Horizontal Displacement: Horizontal sweep, which is normally centered, can be displaced (offset) left or right by any multiple of 1/2 inches.
- (e) Vertical Jog: By means of a manual key-switch, the film is made to move up or down, at either of two speeds, to bring it precisely to a desired location.
- (f) Vertical Displacement: At the beginning of scan the film is automatically moved up or down by a selectable number of 1/8 inches.

Results So Far

A first version of the equipment has been built and tested. The transmitter and receiver each consists of a cabinet, containing most of the electronic components, and an optical-mechanical assembly on its top. They are shown, respectively, in Figures 9 and 10. The mechanical and electronic construction was done in an expansive "breadboard" fashion - to facilitate development work and future modifications. [An operational system could be made appreciably more compact and "streamlined"].

This version has a single resolution of 250 pels/inch, the spot size being about 3X4 mils. The sampling rate is variable, but practically only the slowest rate, with full 8 bits/pel transmission, has been used. None of controls d-f above was built into that version, and the transmission was in bit-serial (mode 1) only, and that over a short shielded cable. Instead of the light-collecting optics, there is a long array of photodiodes, (solar-cell type) facing an opaque white diffuser. This necessarily results in lower signal and higher noise level, as discussed above.

Images of fairly good diagnostic quality have been produced by this system. Owing to the relatively low signal/noise ratio, discriminable source density is limited to 2.5 optical density units. The spatial resolution of the transmitter is commensurate with the sampling density, i.e. 250 to the inch.

The reproduced image on the film shows two defects that are thought to be due to the lack of antihalation backing, namely some diffusion, with a resultant reduction of resolution, and interference between the two film surfaces, causing a visible random fringe pattern. A solution to this problem is sought in cooperation with the film manufacturer. Of course, a normal wet-process film would not exhibit these problems. In addition there appear some horizontal and vertical streaks due to minor defects in the circuitry and in the oven, respectively.

Current and Future Work

Current work is aimed at achieving the full capabilities, as described above, particularly resolution of up to 1000 pels/inch, discriminable black levels of at least 3.5

density units (2.9 at the highest transmission rate), and computer interfacing (mode 2 transmission).

Within the present activity, the first version of the transmitter is being improved and modified, in line with the above, whereby the optical system is essentially rebuilt anew. This equipment is to be loaned to the Radiology Department of Harvard Medical School to be connected to a PDP-11 computer used for research at the Peter Bent Brigham Hospital; it will also be operated experimentally within the hospital as a local radiograph facsimile system for evaluation of potential utility and possible applications.

Concurrently, a second transmitter, in the improved version, is being built for use by the CIPG at M.I.T. Here the optical system is different again, since it is designed around a special lens that permits single-pass scanning. This lens will allow 1000 lines/in. resolution over the entire 14" width of the film.

Near future work will lead to (a) an improved version of the receiver, (b) operational models - for various experimental long-distance transmission systems and other computer installations, (c) evaluation of optimal scan parameters for various source films, (d) proliferation of computer applications (e.g. image enhancement, feature extraction and diagnostic aids, as discussed below). Looking ahead, the need for further improvement in this technology is manifest. At the beginning of the development of inexpensive flat-field scanners, there already existed expensive, high performance curved field systems, mainly developed for military applications. These use post-objective scanning by precise polygonal spinners, low f/ number objectives on-axis, and cylindrical focal surfaces. In order to approach the performance of such systems with flat-field scanners, some additional improvements are needed. One is better objectives capable of operating over scan angles of 30° or so at low f/ numbers. The design of such lenses is simplified somewhat by neglecting geometrical distortion and relying on an optical reference, such as a grating, to position the scanning beam accurately on the page. Likewise the precision of the deflecting device may be greatly reduced by relying on horizontal position sensing and buffering of the video signal.

Possible Applications

The medical X-ray version of the system is expected to be useful primarily for remote diagnosis, consultation, and study. At least some of this at present is done by mail, and it is anticipated that the availability of an inexpensive, high quality system, especially one which does not require parting with the original, would increase this traffic. Although it appears unlikely that large scale computer processing of X-rays such as automatic diagnosis, is likely in the near future, minor processing such as correction of exposure errors without re-exposing the patient might well become practical. Finally, the storage and retrieval area of radiology is most likely to be amenable to technological solutions.

Computer processing of imagery other than X-rays has become almost commonplace. An example is the "Electronic Darkroom" now being developed at M.I.T. for the Associated Press, while of course the work of JPL in the space program is well known. We believe that laser systems such as the one described here will eventually supplant others in this application because of their higher quality and lower cost. Finally there appear to be many applications in the graphic arts. Laser scanner systems for plate making and other purposes are now commercially available. Flat-field scanners are of particular importance because the copy is often rigid. Resolution requirements are of the order of 20,000 picture elements/scan line while the geometrical accuracy desired is very high. It would appear that the techniques discussed in this paper are capable of meeting these requirements economically.

References

1. Schreiber, W.F., "Laser/Dry Silver Recorder", SPIE Seminar, San Diego, August 1974.
Schreiber, W.F., "A Laser/Dry Silver Facsimile System", TAPPI, The Journal of the Technical Association of the Pulp and Paper Industry, Vol. 57, No. 4, April 1974, pp. 91-93.
Schreiber, W.F., "A Laser/Dry Silver Facsimile System", Journal of the Institute of Image Electronics Engineers of Japan, June 1974, pp. 94-101.
2. Schreiber, W.F. and Gronemann, U., "Laser Scanning of Radiographs", N.Y. Academy of Science Third Conference on the Laser, April 25, 1975.
3. Beiser, Leo, "Laser Scanning Systems", Laser Applications-Vol.2, Academic Press, 1974, pp. 53-160.
4. "Device Photolithography", Bell System Technical Journal, November 1970, (entire issue).
5. Troxel, D.E., "Computer Editing of News Photos", Trans. IEEE, Systems, Man, and Cybernetics, to be published Nov., 1975.

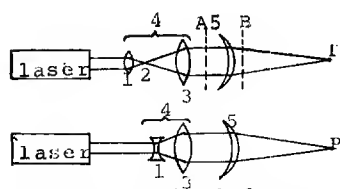


Fig. 1. Typical laser optical systems. 1. Beam expansion lens. 2. Position of spatial filter, if used. 3. Collimating lens. 4. Beam expander. 5. Focussing lens

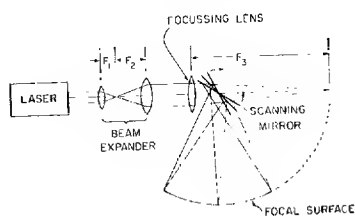


Fig. 2. Post Objective Deflection

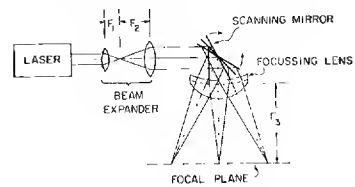


Fig. 3. Pre Objective Deflection

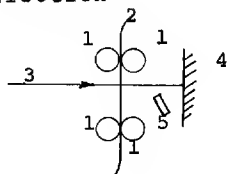


Fig. 6. Original light collection system. 1. Pinch rollers. 2. Film. 3. Laser beam. 4. White reflector. 5. Photo-diode.

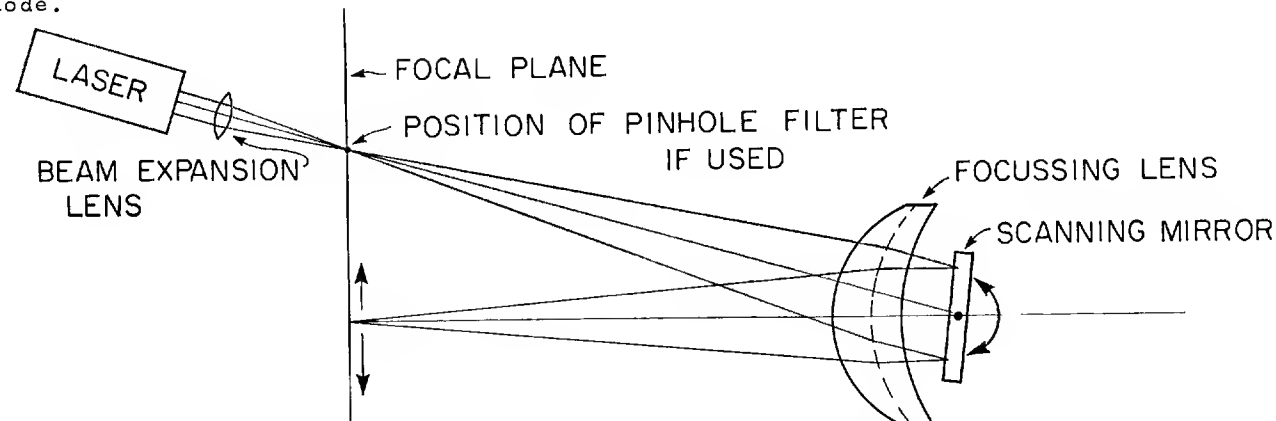


Fig. 4 Double Pass Optical System. Since the beam is collimated at the scanning mirror, the reflected light is in focus over a large area of the focal plane.

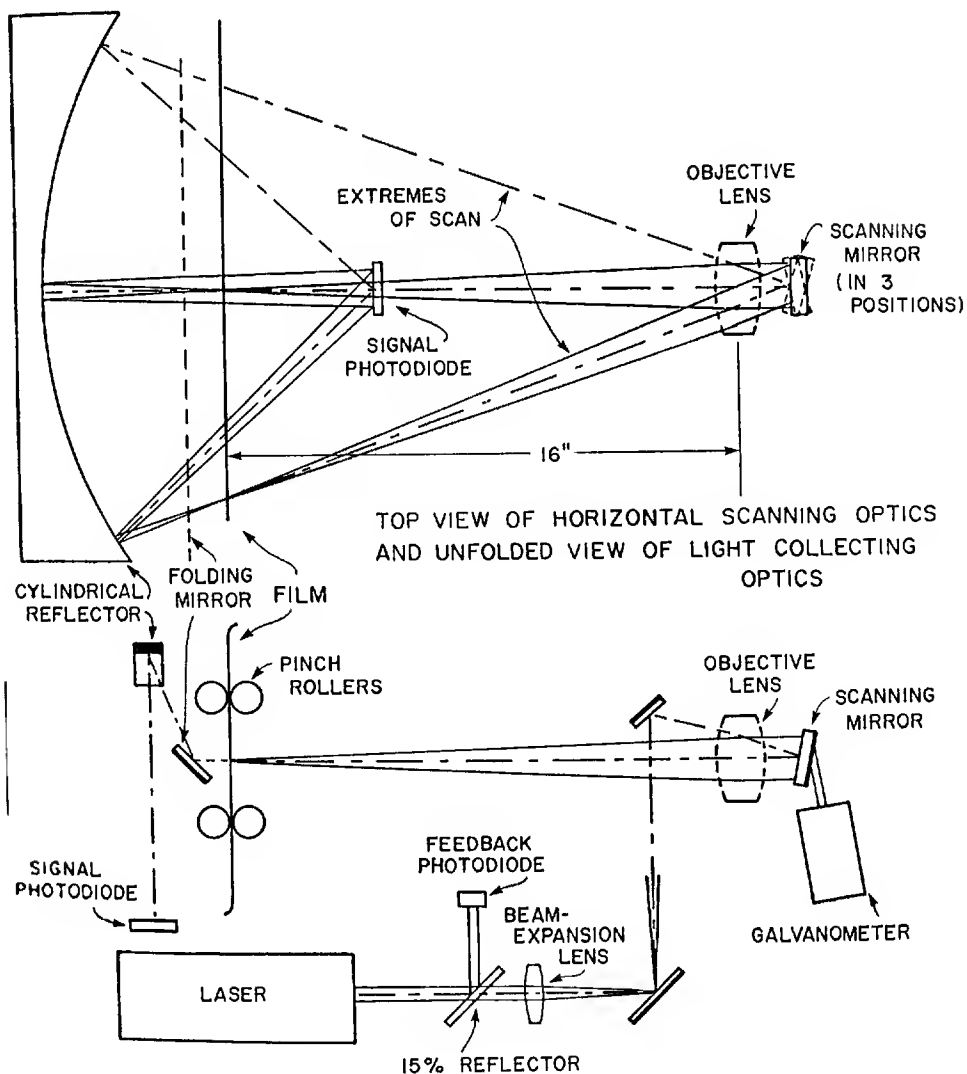
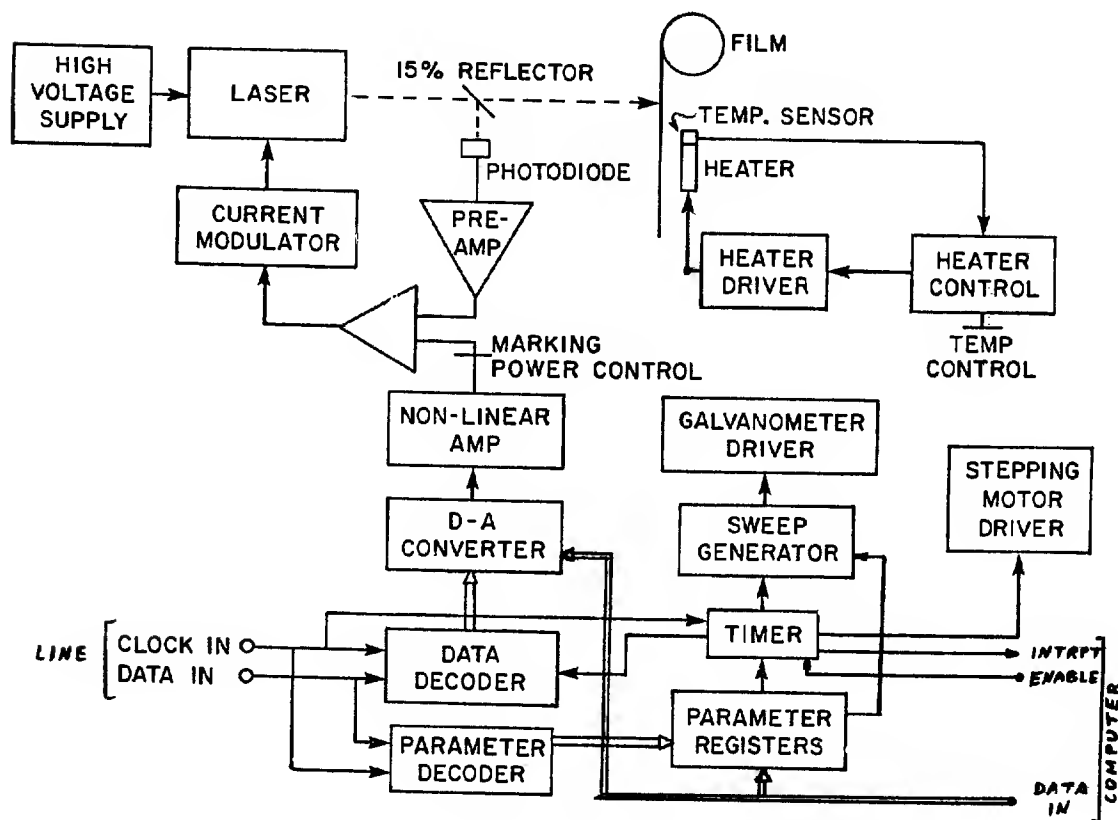
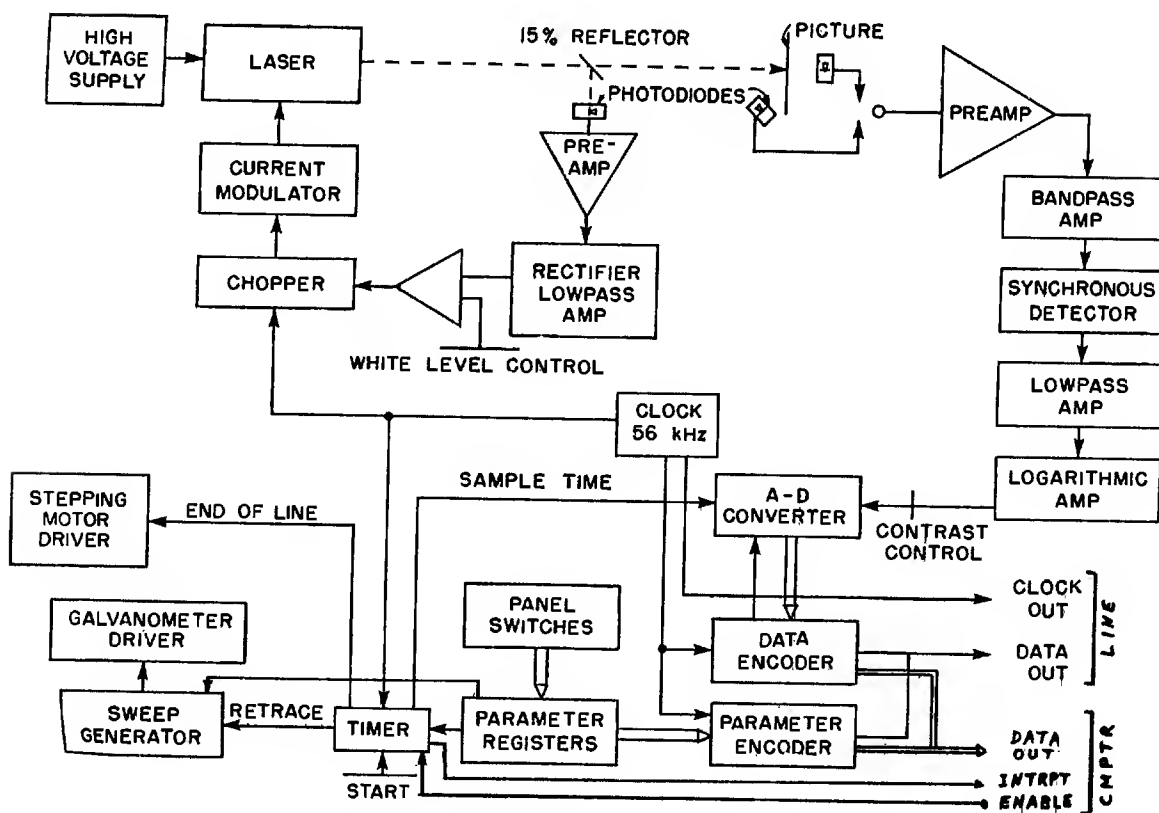


Fig. 5. Optical System and Large Collector



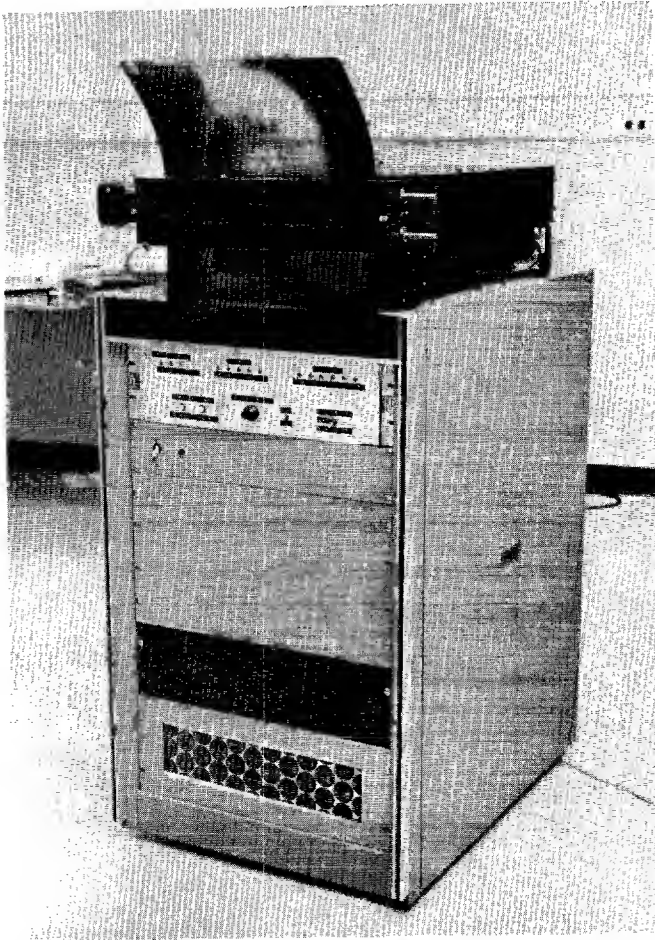


Fig. 9. Transmitter

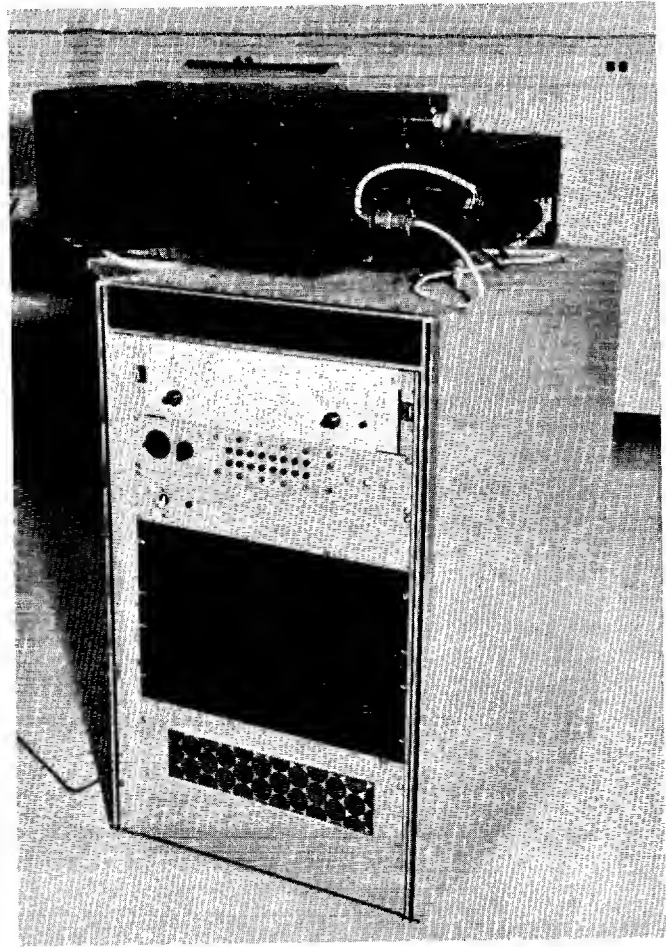


Fig. 10. Receiver

REAL-TIME IMAGE DATA ACQUISITION/RETRIEVAL (RIDAR) SYSTEM*

Harold H. Kantner
CAI, a Division of Bourns, Inc.
Barrington, Illinois

Abstract

Aerial photoreconnaissance with optoelectronic sensors generates video data at formidable rates. All facets of the image data handling and assimilation problem are brought together in presentation of a viable means for its solution. A real-time image data acquisition/retrieval (RIDAR) system is defined and described. System architecture is that of a ground-based data hub which serves operational requirements of reconnaissance and surveillance activities. The RIDAR system functions to enable human beings to screen wideband video signals continuously incoming from high-resolution large-format sensors of remote survey stations as they sweep rapidly over scenes of interest. A basic system feature is its immediate display of a widefield index picture which shifts through the observation field synchronously with sensor transit of the object scene. Pointer designation of an image detail selects its subfield for magnified display and/or speed-print of a high-resolution document. Capture of incoming image data in permanent mass storage permits subsequent playback and selective recall through the same display and document generator system. The proposed RIDAR system can be extended toward a more universal image data processor by incorporating optical and electronic means for parallel computation over two- and three-dimensional fields.

Introduction

Telereconnaissance cameras and finders are capable of generating high resolution video displays at data rates beyond the assimilative abilities of a human observer. However, the screening of photoreconnaissance mission data for visual images of interest to human beings is a task best done by human observers. Their remarkable perceptual faculties deal concurrently with aggregation and discrimination - - integration and differentiation - - over large fields. No automatic data processor can match the gestalts of human beings.

At a receiving terminal the problem of coping with continuous flow of image data at now attainable rates of video teledata is formidable. It will be compounded greatly as sensor resolutions, viewfield coverages and reconnaissance platform transit rates are increased. Scene dynamics complicate the problem further. Tele-reconnaissance image data differ significantly from those of conventional television. An incoming reconnaissance image stream is generally acyclic with little field-to-field repetition of details. Except for special situations and field overlap, the reconnaissance scene changes continuously and quite unpredictably.

In combination with modern tactics, those image characteristics require continuous monitoring of ungapped photodata as they are received during the course of a reconnaissance mission. That calls for an online station which converts incoming broadband video signals into readily assimilable image displays. A viable solution to the data handling and assimilation problem is one which enables effective engagement of human beings with the data flux in its true time scale, and with minimum presentation delay. Such systems are often called real time systems.

Operational Considerations

The image data acquisition/retrieval system described herein is a real time system. It engages an observer with a cinematic event within seconds of its actual occurrence, in contrast to other means which incur the relatively long delays of photographic processing. When equipped with means for addressable mass data storage, it is operable also offline in true or modified time scale. Thus it provides capability for change detection by recursive processing of a cumulative file and for instruction and communication applications.

A primary operational requirement of the ground station image data system is capability for rapid selection of image subfields to be magnified and speed-printed in high-resolution grey-scaled documents for photointerpretation and communication. Direct addressing of high-resolution subfields via a transient,

* Development of system concepts described herein was supported in-house.

wide-field image display index accommodates that function during both online and offline operation. Fixed block data addresses automatically assigned to the incoming video signal facilitate offline image retrieval and file processing, including correlative data annotation.

For reconnaissance purposes, an important aspect of a ground-based data hub is time delay in delivering sensory images to users. Costs which increase with that delay include deterioration of time-critical response opportunities and deterioration of the quality of information which human minds assemble by associating sensor data with those from other sources.

The real time image data acquisition/retrieval (RIDAR) system shown in the block diagram, Figure 1, cuts both of those separate and distinct costs. It does so by display of low-resolution details in widefield image context concurrently with display of selected high-resolution image details. In other words, it exhibits both the "forest" and selected "trees" at resolutions appropriate for each. In that sense, it offers a very crude approximation to the simultaneous foveal/peripheral capability of human vision.

Data Systems Approach

The RIDAR approach presented in Figure 1 is sufficiently general to accommodate wide angle and telescopic telesensors of various types. For example, different display schemes such as the three exhibited as Figures 2, 3 and 5 can be encompassed by a single unified RIDAR system design. Each of the display schemes portrays outputs of the pair of monitor subsystems enclosed by dashed lines in the block diagram, Figure 1.

Two of the display schemes present imagery from long range oblique photography (LOROP) telesensors. In one case, Figure 2, a quasi-panoramic image sequence is obtained from a telescope with focal surface scanning. In the second LOROP example, Figure 3, the video signal is derived from a telescope operated with "pushbroom" scan. The third display scheme, Figure 5, shows application of the RIDAR system for presentation of panoramic imagery from a wide angle real time surveillance (WARTS) system. In that case, seeing the down-translating index display is analogous to observing the terrain beneath an airvehicle framed through a cross-track panoramic viewer during overflight. These examples show that incoming and archival video data from both wide-angle and telescopic sensors can be handled with one real time image data acquisition/retrieval (RIDAR) system.

All major RIDAR subsystem blocks, including the display devices, are diagrammed in Figure 1. Its signal paths show that incoming video data are written concurrently into three separate memory subsystems. The three video stores form an indexed heirarchy of progressively greater capacity and picture field resolution. As in conventional computer technology, a mass storage means such as tape provides the largest capacity for permanent recording of highest resolution video data. Next in order of capacity is the video disc subsystem with more rapid data access. The third memory subsystem, an electronic video store, provides least capacity but fastest access.

The two lower capacity memory modules for temporary data storage employ cyclic erase/write/read progression to provide simultaneous write/read capability. They incorporate triple record/reproduce units to enable cyclic reading on two units while the third is being written. Of the two temporary video stores one, a video disc module, continuously acquires incoming video data at full resolution. The other transient data module incorporates electronic means, such as video scan converters, suitable for high data rate refreshment of a raster display. Continuously incoming video data is written into it at a density which provides about one tenth the linear resolution of the disc system. Thus, the low resolution electronic storage module has about one percent of the two-dimensional data capacity of the high resolution video disc storage module.

Shift Control of Moving Picture Index

Both the disc and the electronic memories continuously capture the same video data at the same time from either the data link or via tape playback. They provide transient storage for the same interval, but at different levels of picture resolution. The write/read devices of each subsystem operate at a cyclic rate proportional to effective ω , the angular rate at which an image point transits the picture field. Conversely, the field of observation and/or access may be said to translate over a specific picture point at a shift rate proportional to effective ω . Accordingly, data in transient storage is accessible only during a shift interval which is inversely proportional to effective ω .

As shown in Figure 1, the ω field shift control signal is extracted from the composite video delivered by the data link or a tape record. Via a controller, it governs the rate at which temporarily stored data is rolled over concurrently in high-and-low resolution transient video subsystems.

When presented on a high performance television monitor, the low resolution recording serves as a wide-field picture index to high-resolution picture subfields stored in the disc module. An address matrix suitably associated with both temporary recordings provides for rapid access to, and transfer of, high resolution subfields during the field shift interval. Slow rollover of the low-resolution data by a read raster synchronized with the TV display raster provides a sliding index display scheme as detailed in Figures 4 and 6 for video data from strip-image and panoramic sensor systems. In accordance with Figures 1, 2, 3 and 5, light pen selection of up to 100 subfields from the moving index enables buffered static display and/or printing of magnified image details.

Indexed Record Library

The high-capacity high-resolution mass data system provides for acquisition, compilation, editing and offline processing of a permanent tape-recorded reconnaissance library. The module of at least two tape transports is linked with a data processor coupled also with an archive module for data tape exchange as shown in Figure 1.

Omission of tape data storage and control is possible if complete data recall and edit capability is unwarranted. In that case, however, only selected subfields of the incoming data can be directed to a copy station for permanent recording. Forced choice due to equipment limitation usually would be hastened by brevity of the decision interval. The non-recurring opportunity for selection of a subfield is limited to the time (usually seconds) required for its transit over an index field which presents as many as 100 alternatives.

Summary

As proposed herein, RIDAR system architecture serves a specific operational requirement. The broad system function is to enable human beings to screen wideband video signals continuously incoming from high-resolution large-format sensors aboard remote survey vehicles as they move rapidly over terrain of interest. A basic system feature is its immediate display of a widefield index picture which shifts through the observation field synchronously with sensor transit of the object scene. Pointer designation of an image detail selects its subfield for magnified display and/or speed-print of a high-resolution document. Capture of incoming image data in permanent mass storage permits subsequent playback and selective recall through the same display and document generator system. The proposed RIDAR system can be extended toward a more universal image data processor by incorporating optical and electronic means for parallel computation over two- and three-dimensional fields.

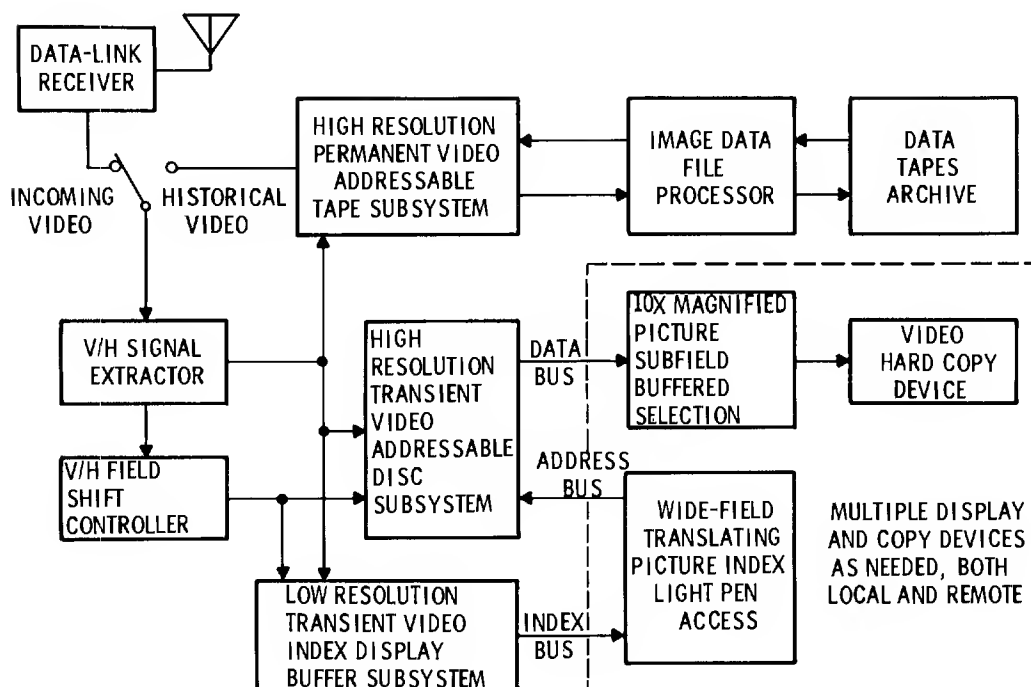


FIGURE 1. REALTIME IMAGE DATA ACQUISITION/RETRIEVAL (RIDAR) SYSTEM

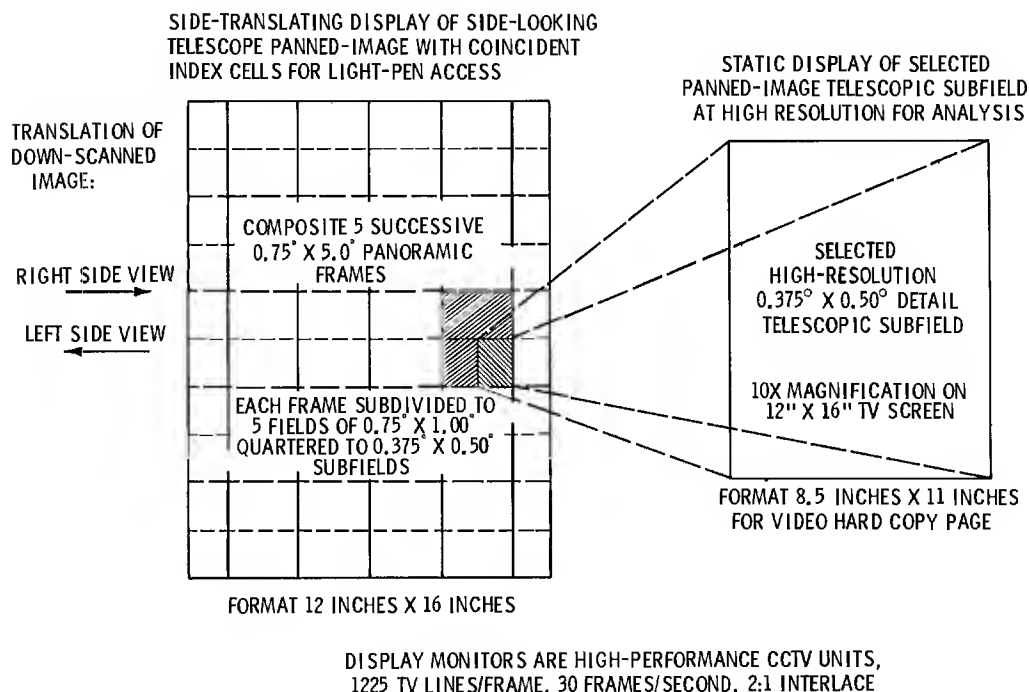
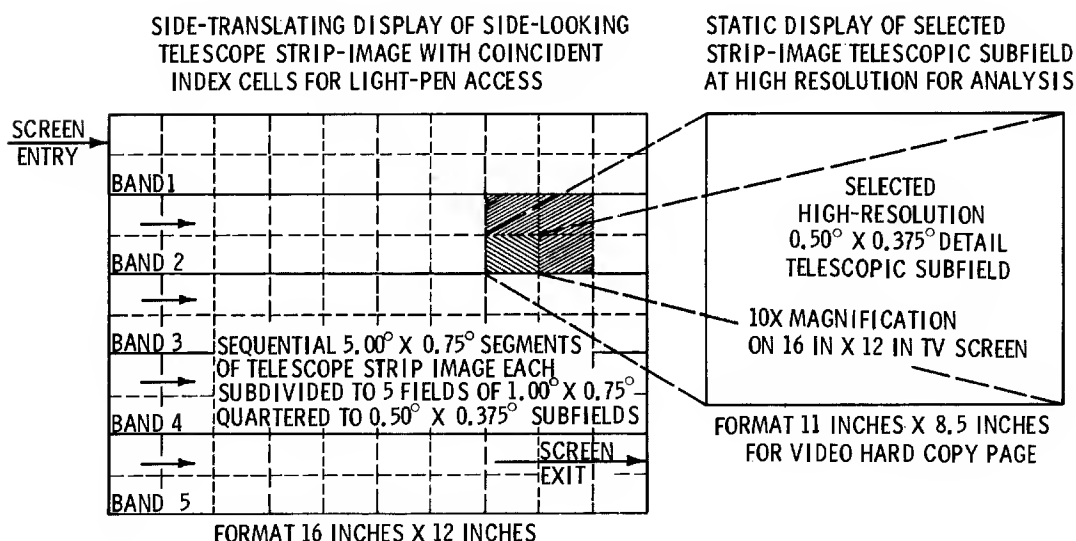


FIGURE 2. WIDE AND NARROW-FIELD VIDEO DISPLAY OF DOWN-SCANNED FOCAL SURFACE IN LOROP



DISPLAY MONITORS ARE HIGH-PERFORMANCE CCTV UNITS, 1225 TV LINES/FRAME, 30 FRAMES/SECOND, 2:1 INTERLACE

FIGURE 3. WIDE AND NARROW-FIELD STRIP-IMAGE VIDEO DISPLAYS FOR LOROP

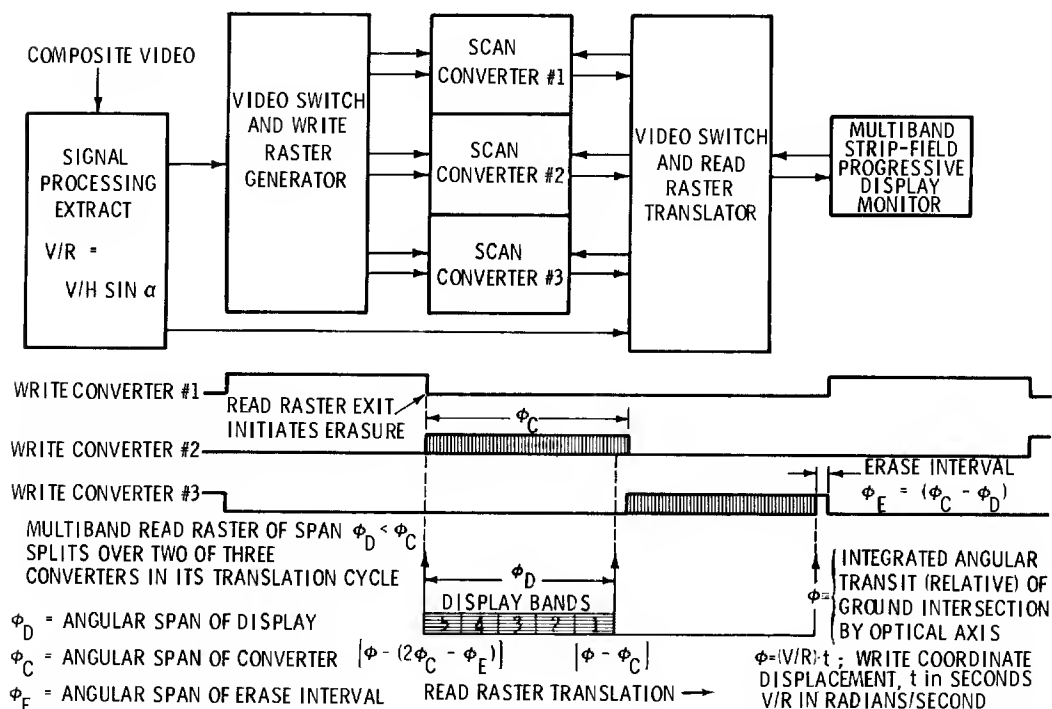


FIGURE 4. STRIP-IMAGE SUBFIELD SEQUENCE SCAN CONVERTER



AN INVESTIGATION OF MSE CONTRIBUTIONS IN TRANSFORM IMAGE CODING SCHEMES

John R. Parsons *
and
Andrew G. Tescher *

Abstract

The mean square error (MSE) is a classical measure of image distortion. However, this metric is generally not a faithful indication of subjective image quality. We attempt to correct this deficiency by addressing the individual error sources in transform image coding. Specifically, the component of the MSE introduced by transform coefficient deletion is separated from requantization effects. Results are demonstrated in both numerical and pictorial form.

Introduction

Efficient transmission of pictorial data is of primary importance in various fields. For an ever-increasing number of diverse systems, better than television quality pictorial information is required to be sent at high picture rates over channels of limited bandwidth. Common to all image coding applications is the maintenance of fidelity of the decoded image. Recognizing the system-specific nature of quality requirements, one could base a coder design solely on the results of exhaustive simulations and subjective image evaluations for a wide range of image types and coder parameters. For a transform image coding scheme the latter may include quantization, bit assignments, transform type, sub-block size, coefficient selection, channel error effects and adaptivity. Therefore a quantitative metric correlating with subjective assessments can play a vital role in making tradeoffs between image quality and coder parameters in that fewer subjective evaluations are required and the parameters can perhaps be modelled in their impact on such a metric.

The mean square error (MSE) has been a traditional metric employed for image error analysis because of its classical application in one-dimensional signal analysis.⁽¹⁾ Unfortunately its correlation with subjective rankings of image quality is not entirely satisfactory. It is the purpose of this paper to investigate a refinement of the MSE criterion as it applies to imagery which has been subjected to various transform coding schemes. In the following sections we discuss the basic components of the transform coder concept and the traditional role of the MSE. Then we propose a decomposition of the MSE and describe an experiment to investigate this refined metric and its correlation with subjective evaluation. Finally, conclusions as to the applicability of such a method are addressed.

Transform Image Coding

Transform image coding has been treated by a number of authors^(2,3,4) as a viable means of transmitting digital image data in a compressed form. In this section we will discuss the basic elements and concepts of such a coding technique. Consider an image I discretely sampled over an $N \times N$ grid where each sampled brightness value $I(i,j)$ is quantized to L bits. In this case $i,j = 1,2,\dots,N$ and each brightness value (pixel) can take on one of 2^L assignments which, except for a possible scaling factor, fall between 0 and $2^L - 1$.

Direct coding of pixel values results in a sub-optimum coder since image correlation is not reduced. Various two-dimensional linear transformations have been considered to map the pixel values into a space where the coefficients are approximately uncorrelated and the maximum image energy is compacted into the low order terms. Fourier, Cosine, Slant and Hadamard transforms have been considered for this application.

Figure 1 shows the model we will be using for the source coder/decoder problem and Figure 2 is a pictorial representation of the various steps. The array $[I]_N$ of image data is presented first to a partitioning operation where the larger $N \times N$ array is segmented into smaller $n \times n$ arrays $[I]_n$. N might be several thousand while n is typically less than 100. This partitioning is introduced for a variety of reasons, among them considerations of the correlation properties of the image since decorrelation of the data over a region larger than its correlation size is not needed. Additional consideration is the implementation of the transform algorithm with a minimum of computational complexity. Also, partitioning allows the coder to be sensitive to varying scene statistics and forms the basis for a large class of adaptive techniques.

The $n \times n$ partitioned data at L bits is then transformed into an $n \times n$ array of L bit coefficients $C(k,l)$ for $k,l = 1,2,\dots,n$ by the linear transform operator T . In matrix notation the transform coefficient array $[C]_n$ is given by:

$$[C]_n = [T]_n [I]_n [T]_n^t \quad (1)$$

where t denotes a transposed complex conjugate operation. The matrix $[T]_n$ has the elements:

* The authors are with the Aerospace Corporation, El Segundo, California

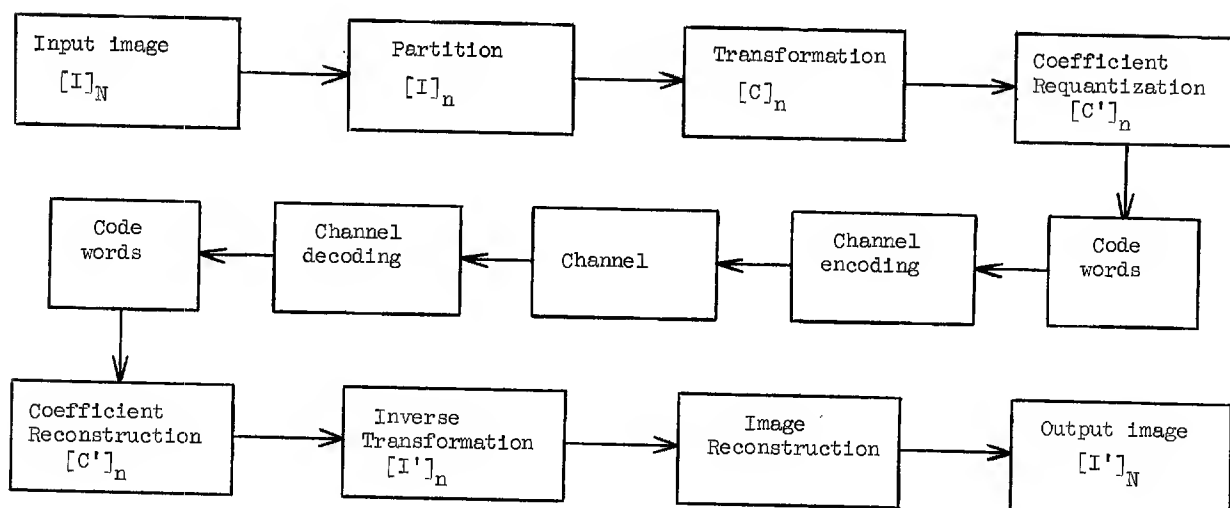


FIGURE 1. Transform Image Coding Communication Model

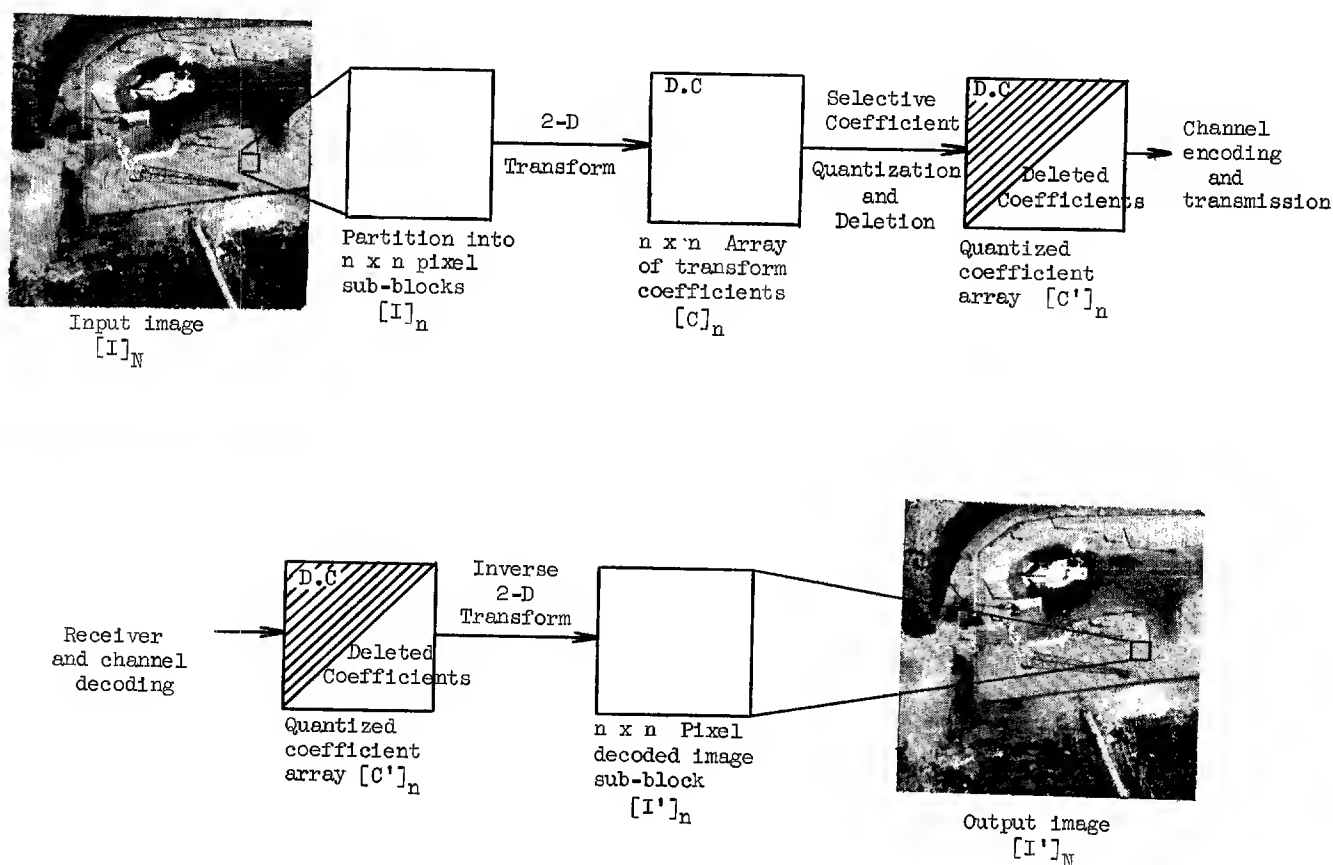


FIGURE 2. Pictorial representation of transform image coding model

Fourier Transform

$$T(k, \ell) = \frac{1}{n} \exp[-2\pi i k \ell / n] \quad (2)$$

Cosine Transform⁽⁵⁾

$$T(k, \ell) = \frac{1}{n} \cos[(2k+\ell)\ell\pi/2n] \cdot \cos[(2\ell+1)k\pi/2n] \quad (3)$$

Hadamard Transform⁽⁶⁾

$$T(k, \ell) = \frac{1}{n} \prod_{r=0}^{n-1} \text{sgn}[\cos^{\ell_r} (2^{r-1} \pi (k+1))] \quad (4)$$

where

$$\text{sgn}(a) = \begin{cases} +1 & a > 0 \\ -1 & a < 0 \end{cases}$$

and ℓ is expressed in binary form as the set

$$\ell = (\ell_{m-1} \ell_{m-2} \dots \ell_1 \ell_0) \text{ with } \ell_r \in (0, 1)$$

Slant Transform⁽⁴⁾

For $n = 4$ we have:

$$[T]_4 = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/5 & 1/5 & -1/5 & -3/5 \\ 1 & -1 & -1 & 1 \\ 1/5 & -3/5 & 3/5 & -1/5 \end{bmatrix} \quad (5)$$

Larger arrays are constructed iteratively.

By design $n \in 2^r$ to allow implementation in a "fast" algorithm. The resulting set of coefficients $[C]_n$ representing the partitioned image data must now be requantized into a coefficient set $[C']_n$ according to some quantization model to produce the desired bit rate reduction. It is this step which is perhaps the most important in any compression algorithm. For example one can perform "zonal" coding by partitioning the transform array and assigning a bit rate to each coefficient within a partition based on rate distortion theory (7) or pre-assigned rates based on subjective evaluations of the decoded imagery. Another strategy is to threshold the coefficients and quantize only those elements greater than the threshold value.

Utilization of adaptive techniques can also be applied. For example, bit assignments can be varied as a function of image statistics over the partition. For this case, the output bits should be buffered to insure a constant bit rate for transmission.

The code word bit stream representing the sequential source coding of each image partition, along with any overhead bits required for the decoding process such as utilized in certain adaptive schemes, is then channel encoded and transmitted. At the receiver the inverse operations are performed and the coefficient set $[C']_n$ reconstructed. The inverse transform operation

$$[I']_n = [T]_n^t [C']_n [T]_n \quad (6)$$

then yields an image partition $[I']_n$ which should be a faithful reproduction of $[I]_n$. These partitions are then used to form the composite reconstructed image $[I']_N$.

Before going on to discuss the comparison of I and I' , consider another operation which can be accomplished before source decoding. The process discussed above can be related to a low pass filtering operation (assuming that the coefficient requantization effects can be ignored) and therefore one can investigate the possibility of applying a high frequency boost or other transfer function compensation to the coefficients of $[C']_n$ for each partition before inverse transformation. Such an action would, for the case of a cosine transform, correspond to the block-mode filtering of digital imagery⁽⁸⁾ except that there is no overlap between partitions to reduce edge effects. Thus one would form the scalar multiplication:

$$[C']_n = F_n[C']_n \quad (7)$$

where F_n is an array of coefficient weights. The utility of such a technique is demonstrated in Figure 3. The original image (a) is digitized at 10 bits/pixel and $N = 512$ while (b) is the reconstructed image after a compression to 2.6 bits/pixel using the cosine transform with $n = 16$. Figure 3(c) is the reconstructed image where the coefficient weighting array $F(k, \ell) = 1 + \sqrt{(k-1)^2 + (\ell-1)^2}/15$ for $k, \ell = 1, 2, \dots, 16$ was first applied to the quantized coefficients of each partition before reconstruction. A radial section of this 2:1 boost filter is shown in 3(d). The improvement is obvious and in fact the filtered image has been subjectively rated superior to the original.

The MSE As A Quality Metric

The most widely used measure of image quality for coding analysis is the normalized mean square error (MSE) between the coder input (I) and output (I') images.

$$\begin{aligned} \text{MSE} &= \sum_{i,j}^N |I(i,j) - I'(i,j)|^2 / \sum_{i,j}^N |I(i,j)|^2 \\ &= \sum_p \left\{ \sum_{i,j}^n |I(i,j) - I'(i,j)|^2 \right\} / \sum_p \left\{ \sum_{i,j}^n |I(i,j)|^2 \right\} \end{aligned} \quad (8)$$

where \sum_p indicates a sum over all partitions.

The orthonormality of the transform [T] allows us to restate (8) as:

$$\text{MSE} = \sum_p \left\{ \sum_{k,\ell}^n |C(k,\ell) - C'(k,\ell)|^2 \right\} / \sum_p \left\{ \sum_{k,\ell}^n |C(k,\ell)|^2 \right\} \quad (9)$$

One can therefore relate the overall MSE to the errors introduced on the coefficients $C(k, \ell)$ by their requantization and/or deletion. The problem of optimizing the quantization algorithm relative to the MSE and subjective quality assessment has been considered previously.⁽⁹⁾ We have found that decoded image errors due to coefficient requantization in zonal coding schemes are more objectional than the smoothing effect introduced by high frequency coefficient deletion (low pass filtering). To develop a useful requantization design strategy which is consistent with subjective quality measures, we separate the two major components of the MSE:

$$\text{MSE} = \text{MSE}_q + \text{MSE}_f \quad (10)$$

where:

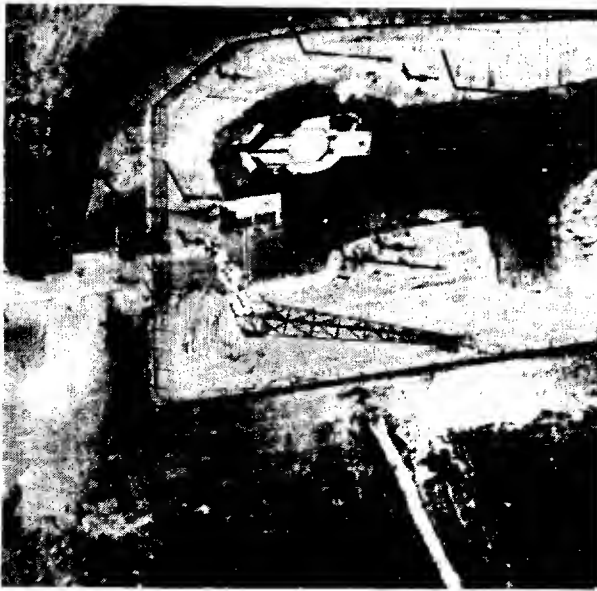
$$\text{MSE}_f = \sum_p \left\{ \sum_{k_d, \ell_d} |C(k_d, \ell_d)|^2 \right\} / \sum_p \left\{ \sum_{k, \ell}^n |C(k, \ell)|^2 \right\}$$

for deleted coefficients $C(k_d, \ell_d)$ is the error due to filtered or deleted coefficients and

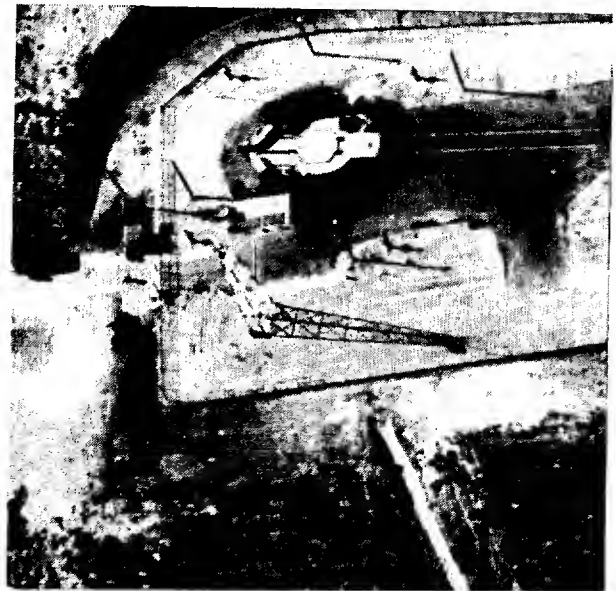
$$\begin{aligned} \text{MSE}_q &= \text{MSE} - \text{MSE}_f \\ &= \sum_p \left\{ \sum_{k, \ell}^n |C(k, \ell) - C'(k, \ell)|^2 \right. \\ &\quad \left. - \sum_{k_d, \ell_d} |C(k_d, \ell_d)|^2 \right\} / \sum_p \left\{ \sum_{k, \ell}^n |C(k, \ell)|^2 \right\} \end{aligned} \quad (11)$$

is the error due to coefficient requantization.

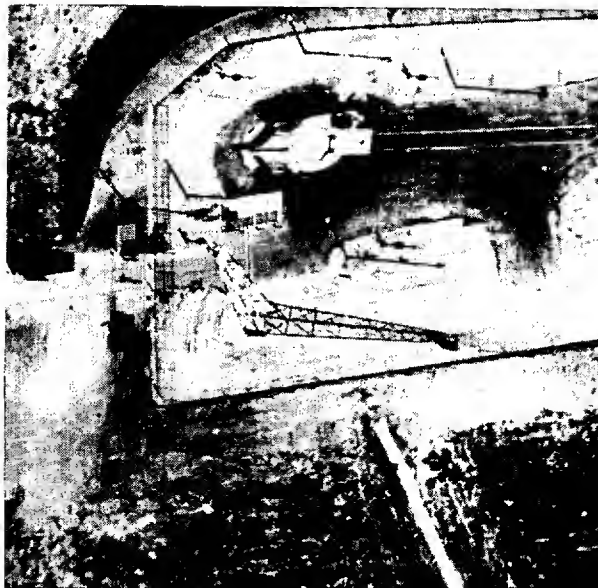
The subjective effects are considerably different for the two error terms. The coefficient requantization generates a "noise-like" distortion and artifacts while deletion of the high frequency terms actually reduces noise, although at the expense of resolution. An obvious observation is that both quantization and filtering effects are undesirable yet unavoidable parts of the compression process, and it is felt that the compression algorithm designer can utilize MSE_f and MSE_q in order to optimize his algorithm. Assuming an optimum quantization procedure is utilized for individual coefficients, a good statistical model is available for transform domain variances, the transmission rate is fixed, and there are a total of $K = n^2$



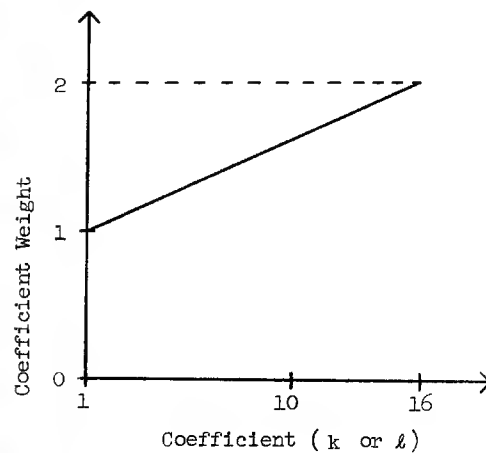
Original (I) image at 10 Bits/Pixel



Reconstructed decoded (I') image after compression to 2.6 Bits/Pixel



Reconstructed decoded image with high frequency boost over each partition



Radial section of high frequency boost filter

FIGURE 3. Compression/Filtering Experiment Using Cosine Transform and 16 x 16 Partitioning

coefficients per sub-block of which K_q are quantized and K_f are deleted (so $K = K_f + K_q$), then the following procedure can be used:

1. Set K_f such that $MSE_q \approx MSE_f$
2. Compare I and I'. If I' contains negligible noise, iteratively decrease K_f .
3. If I' contains excessive additional noise and/or artifacting, iteratively increase K_f .

Step (1) should result in a minimum MSE but steps (2) and (3) can result in improved visual representation by the tradeoff of quantization noise and resolution loss.

Experiment

A computer program has been developed for the simulation of various transform coding methods and it has been used to generate data for actual imagery. Figure 4 illustrates the manner in which this package was utilized in an experiment to evaluate the correlation of MSE_f and MSE_q with subjective image ranking.

The image used for this study is shown in Figure 3(a) and consists of 512^2 ($N=512$) pixel values, each quantized to 10 bits. Analysis of this array indicates that there are 9 bits of useful image data and furthermore its structure is representative of what might be encountered in a low altitude remote sensing application. This image was partitioned into one of two sizes ($n = 8$ or 16) and one of 3 transforms applied (Cosine, Slant, Hadamard). Selected coefficients were then requantized and others deleted to give bit rates (R) of .64, 1.4, 2.6, 4.6 bits/pixel for $n = 16$ and rates of 1.28, 2.6, 4.3, 6.9 for $n = 8$. The requantization scheme consisted of first looking at the coefficients in bands parallel to the opposing diagonal of $[C]_n$ (Figure 5(a)). We, and others(11), have observed that iso-variance regions of $\sum_p [C]_n$ tend to line up quite well with this zonal approach. After determining the coefficient variance σ_z^2 for the z th zone one of the following two methods was applied.

1. The maximum absolute value $|C_{\max}|$ in each band was determined and the coefficients in that zone uniformly requantized between $\pm |C_{\max}| \cdot C_{\max}$ was quantized to 10 bits for each zone.
2. The variance σ_z^2 was used to define a gaussian quantizer for that band. σ_z^2 was quantized to 10 bits for each zone.

The actual bit assignments for each zone to give the above bit rates are shown in Table 1. In addition to these zonal coding experiments, an adaptive approach as used in reference 10 was utilized for the cosine transform with $n = 16$. In this case the coefficients of $[C]_n$ were first ordered into a one-dimensional sequence of 256 values. The analog for $n = 8$ is shown in Figure 5(b). The coefficients $C(m)$ for $m = 1$ to 256 were then recursively coded using a variance estimate $\hat{\sigma}_m^2$ based on previously coded coefficients, and a gaussian quantizer. The coded coefficients $\hat{C}(m)$ are determined by the quantization operator Q such that $\hat{C}(m) = Q(\frac{\hat{\sigma}_m^2}{\sigma_m^2}, C(m))$ and the next estimated variance is $\hat{\sigma}_{m+1}^2 = W\hat{\sigma}_m^2 + (1-W)\hat{C}(m)^2$ for a pre-determined weight W . The primary assumptions of such a technique are that the $C(m)$ are uncorrelated and the σ_m^2 are highly correlated. This particular coder was operated at 1.55 bits/pixel.

Each of the above 49 coded data sets was decoded and the resulting images $[I']_N$ were both numerically and subjectively compared to $[I]_N$. The numerical comparison consisted of a computation of MSE , MSE_f and MSE_q for each image $[I']_N$ while the subjective analysis consisted of a ranking of the displayed imagery according to a variety of tasks aimed at "exploiting" the imagery. Care was taken to insure that the subjective ranking was somehow tied to how the data would be used.

The displayed imagery represented a wide range of quality and bit rates as demonstrated by Figures 6 and 7. Figure 6(a) is the decoded image for the cosine transform with $n = 16$, $R = 2.6$ bits and uniform requantization while 6(c) is the same transform with $R = .64$ bits. The corresponding scaled absolute value difference images ($\alpha|[I]_N - [I']_N|$) are shown in Figures 6(b) and 6(d). The scaling factor α was set to 15 to enhance the visual impact of the error distribution. One can immediately note the correlation of errors with image edges and other regions of high spatial frequency content. Since image information may be located at or near edges, it is a natural step to look at the performance of the adaptive scheme. It should be sensitive to the frequency content (or sub-block "business") and therefore tend to distribute the errors over the entire scene rather than at edges. Figure 7 demonstrates that phenomenon. Figure 7(a) is the decoded image for $R = 1.4$ bits, $n = 16$ and gaussian requantization of the cosine transform coefficients, while 7(b) is the scaled ($\alpha = 15$) difference image. Figure 7(c) is the decoded image for the adaptive case ($R = 1.55$) with 7(d) the scaled difference image. Comparison of 7(b) and 7(d) illustrates the superior performance of the adaptive approach in decorrelating coding errors and image structure, while providing an overall MSE roughly half that of the non-adaptive case for nearly equal bit rates.

The results of the computed objective (MSE) metrics are shown in Table 2 for all 49 cases used in the experiment. The cosine transform is clearly the superior performer when one compares the metrics for equivalent quantizer, R and n . Also the gaussian quantizer is superior to the uniform case, but the margin is surprisingly narrow. Finally, the apparent ambiguity in MSE_f and MSE_q between Slant and Hadamard transform cases should be noted. This is due to the different convergence properties of the higher frequency Slant and Hadamard terms.

A regression analysis was performed on each ranking of the image data and the corresponding set of metrics, this being done for all such rankings. Table 2 shows the resulting correlation coefficients for each MSE metric.

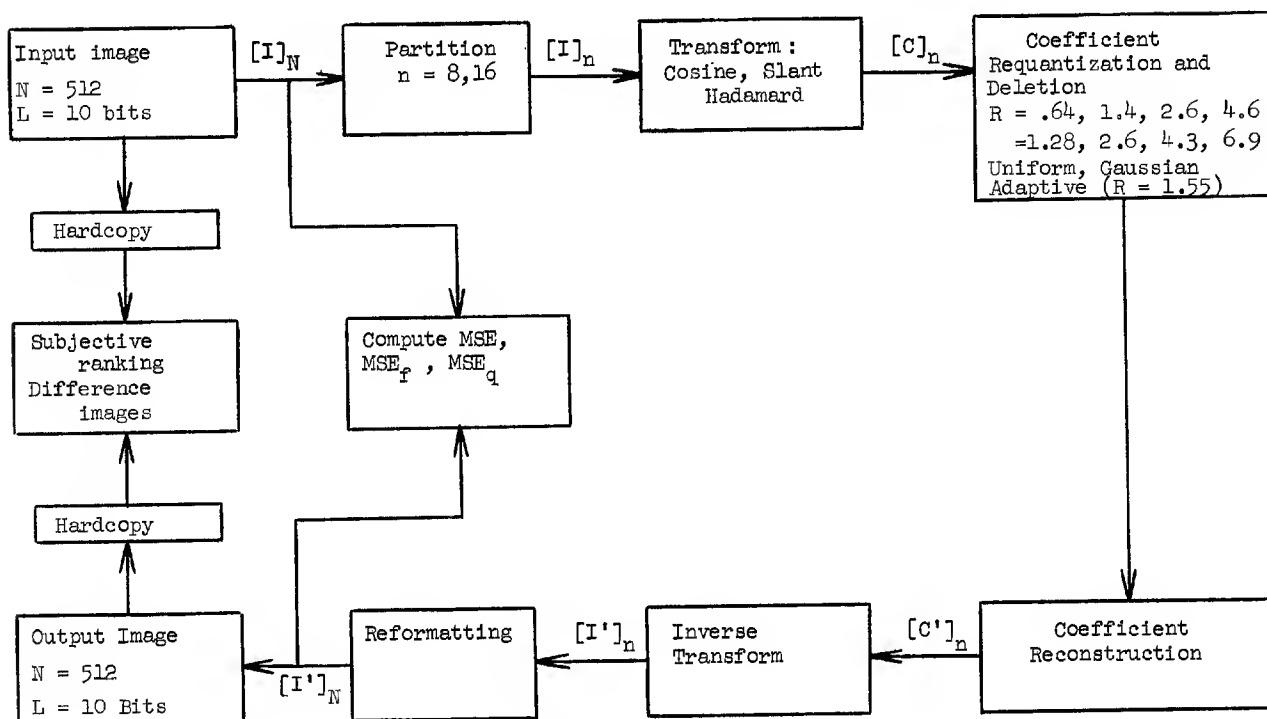


FIGURE 4. Transform Coding Evaluation Experiment

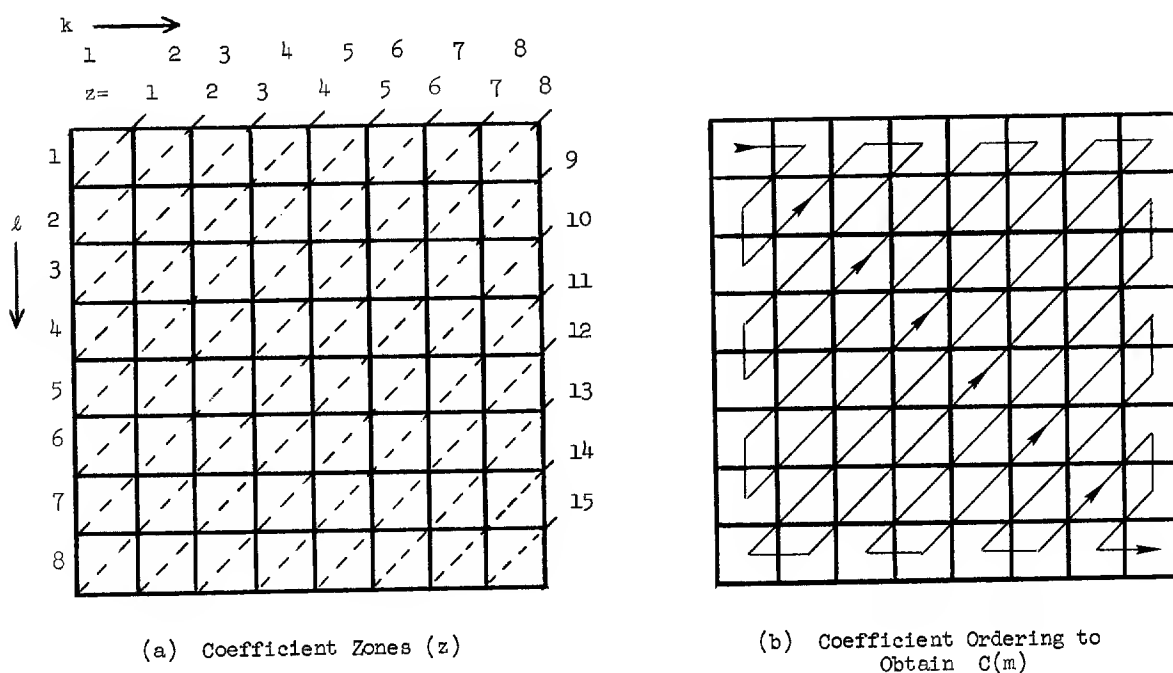
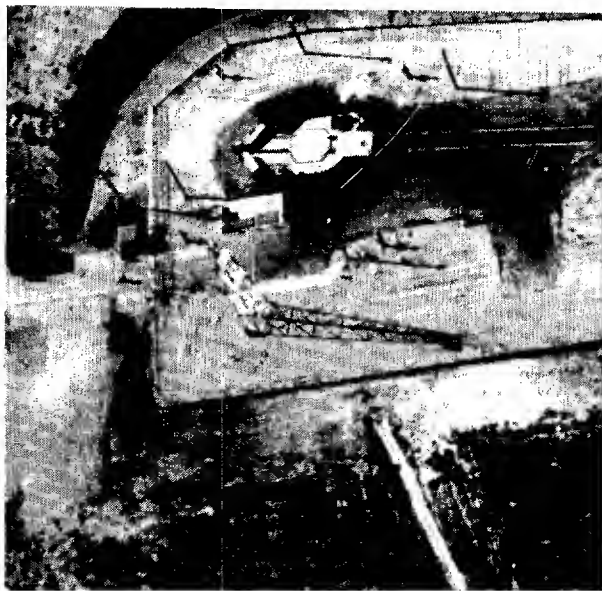


FIGURE 5. Definition of Transform Domain Zones and Coefficient Ordering for 8 x 8 Partitioning



(a)

Reconstructed decoded (I') image for $n = 16$,
 $R = 2.6$ Bits/Pixel, $MSE = .073\%$



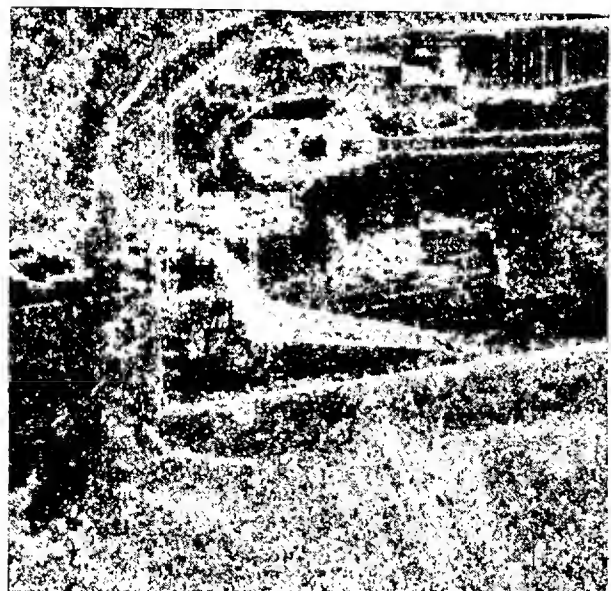
(b)

Difference image $\alpha|I-I'|$ for case $R = 2.6$
 and $\alpha = 15$



(c)

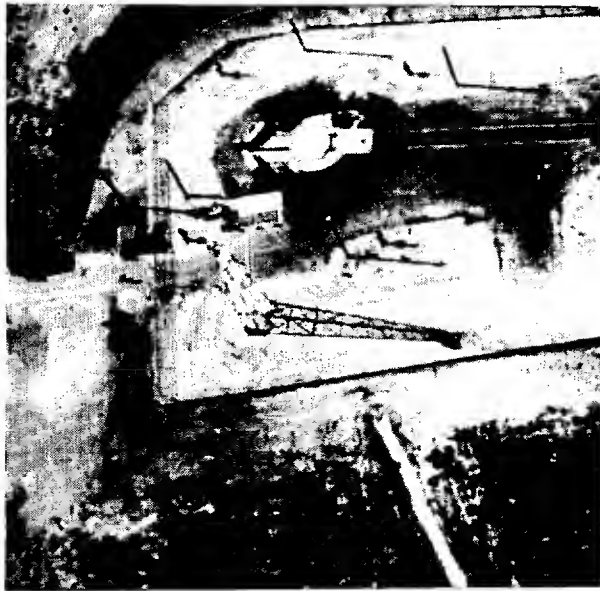
Reconstructed decoded (I') image for $n = 16$,
 $R = .64$ Bits/Pixel, $MSE = .59\%$



(d)

Difference image $\alpha|I-I'|$ for $R = .64$ and
 $\alpha = 15$

FIGURE 6. Cosine Transform Coding Examples with Uniform Quantizer



(a)

Reconstructed decoded (I') image for $n = 16$,
 $R = 1.4$ Bits/Pixel, Gaussian Quantizer,
 $MSE = .17\%$



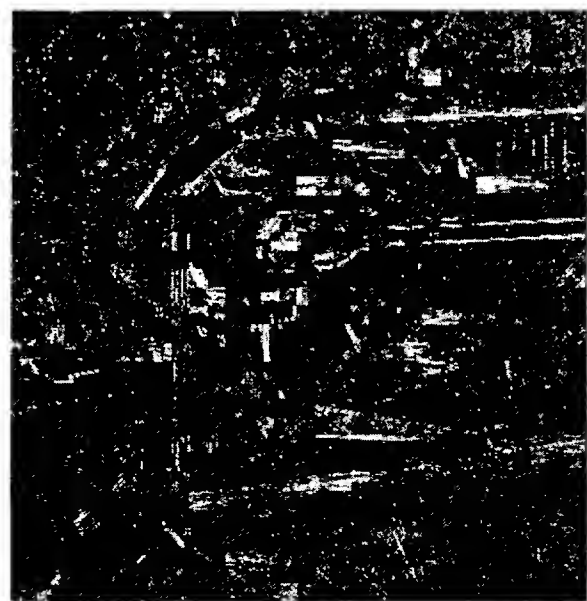
(b)

Difference Image $\alpha|I-I'|$ for case $R = 1.4$ and
 $\alpha = 15$



(c)

Reconstructed decoded (I') image for adaptive
 coder, $n = 16$, $R = 1.55$ Bits/Pixel, $MSE = .08\%$



(d)

Difference image $\alpha|I-I'|$ for adaptive case and
 $\alpha = 15$

FIGURE 7. Comparison of Adaptive and Non-Adaptive Coders using Cosine Transform

		zone(z)																												
n	R	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17 to 31												
16	0.64	10	9	8	8	6	DELETED																							
	1.4	10	9	8	8	6															4	4	4	3	3					
	2.6	10	9	8	8	6															4	4	4	3	3	3	3	3	3	3
	4.6	10	9	8	8	6															4	4	4	3	3	3	3	3	3	3
8	1.28	10	9	8	DELETED														3											
	2.6	10	9	8																	8	6								
	4.3	10	9	8																	8	6	4	4	4					
	6.9	10	9	8																	8	6	4	4	4	3	3	3	3	3

TABLE 1. Zone Bit Assignments for Experiment Bit Rates (R) and Partition Size (n).

n	R	QUANTIZER	TRANSFORM								
			COSINE			SLANT			HADAMARD		
			MSE	MSE _f	MSE _g	MSE	MSE _f	MSE _g	MSE	MSE _f	MSE _g
16	.64	Uniform	.59	.579	.011	.67	.657	.013	.69	.676	.014
	1.4	"	.19	.17	.020	.26	.23	.03	.28	.25	.03
	2.6	"	.073	.046	.027	.096	.060	.036	.102	.068	.034
	4.6	"	.030	0	.030	.040	0	.040	.036	0	.036
16	.64	Gaussian	.55	.542	.008	.62	.609	.011	.63	.618	.012
	1.4	"	.17	.155	.014	.24	.219	.021	.25	.231	.019
	2.6	"	.070	.049	.021	.09	.062	.028	.088	.062	.026
	4.6	"	.028	.0	.028	.038	0	.038	.035	0	.035
8	1.28	Uniform	.47	.465	.005	.49	.483	.007	.50	.493	.007
	2.6	"	.123	.112	.011	.16	.138	.022	.17	.149	.021
	4.3	"	.047	.023	.024	.060	.030	.030	.061	.033	.028
	6.9	"	.028	0	.028	.030	0	.030	.028	0	.028
8	1.28	Gaussian	.45	.446	.004	.46	.454	.006	.46	.455	.005
	2.6	"	.10	.092	.008	.11	.095	.015	.12	.109	.011
	4.3	"	.03	.018	.012	.040	.020	.020	.038	.020	.018
	6.9	"	.021	0	.021	.024	0	.024	.022	0	.022
	1.55	Adaptive	.080	.046	.034						

TABLE 2. Objective Metrics for Coder Evaluation Experiment

METRIC	CORRELATION COEFFICIENT
MSE	.73
MSE _f	.75
MSE _q	.86

TABLE 3. Objective MSE Metrics and their Correlation with Subjective Image Ranking

Conclusions

The development of a viable and general objective measure of image quality is a difficult task. One should try and identify the salient features of a particular image chain model and incorporate them in some fashion into a particular quality criterion well-suited and in harmony with the image and its intended use. We have taken such a track in the development of a refined MSE error as it applies to the judgement of imagery which has been transform coded. The noise introduced by coefficient requantization (MSE_q) has been found to be a superior measure of quality than the traditional overall MSE. It is hoped that future work in this area will yield still better metrics and aid in the efficient modelling of such data transmission systems.

Acknowledgements

The authors wish to acknowledge the programming support provided by Mr. John N. Hamilton.

References

1. A. G. Tescher and J. R. Parsons, "Cross-Spectrum Error Criterion as an Image Quality Measure," *Applied Optics*, Vol. 12, June 1974, p. 1460-1465.
2. P. A. Wintz, "Transform Picture Coding," *Proc. IEEE*, Vol. 60, No. 70, July, 1972.
3. W. K. Pratt, J. Kane and H. C. Andrews, "Hadamard Transform Image Coding," *Proc. IEEE*, Vol. 57, Jan. 1969.
4. W. K. Pratt, W. Chen and L. R. Welch, "Slant Transform Image Coding," *IEEE Trans. Comm.*, Vol. COM-22, No. 8, Aug. 1974.
5. N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform," *IEEE Trans. Computers*, Vol. C-23, pp. 90-93, Jan. 1974.
6. C. Cardot, "A New Eye on Walsh Functions," *Proc. 1972 Symposium on Applications of Walsh Functions*, AD-744-650, pp. 393-400.
7. L. D. Davisson, "Rate Distortion Theory and Applications," *Proc. IEEE*, Vol. 60, July 1972.
8. B. R. Hunt, "Block-Mode Digital Filtering of Pictures," *Mathematical Biosciences*, Vol. 11 (1971), p. 343-354.
9. M. Tasto and P. A. Wintz, "Image Coding by Adaptive Block Quantization," *IEEE Trans. Comm.*, Vol. com-19, pp. 957-972, Dec. 1971.
10. A. G. Tescher, "Adaptive Intra-and Inter-Frame Image Coding Utilizing Fourier Techniques," *USC Image Processing Research Semiannual Technical Report No. USCEE 444*, pp. 24-33.
11. W. B. Schaming, "Digital Image Transform Encoding," *RCA Corporate Engineering Services, Report RE-20-1-10*, 1973.

A COMPARISON OF HARDWARE IMPLEMENTATIONS OF
THE HADAMARD TRANSFORM FOR REAL-TIME IMAGE CODING

Stephen C. Noble
Air Force Avionics Laboratory
Wright-Patterson Air Force Base, Ohio 45433

Abstract

Several different hardware configurations for implementing the Hadamard transform in real-time are discussed. The discussion is referenced to a 64 point (8 x 8) transform that is used in image coding for bandwidth reduction. A successive implementation of two 8 point transformations with intermediate random access memory using the straight forward N^2 computation is compared to implementations of the Fast Hadamard Transform (FHT). The speed of computing transform coefficients is compared to the hardware requirements of the various techniques. The relationship between computational speed and the resolution and frame rate of television imagery that may be transform encoded using the different methods is illustrated. It is shown that for picture element (pixel) rates up to 5 MHz the N^2 computation is the more economical to implement.

Introduction

Transform coding for the efficient transmission of images has been a subject of interest in recent years. In a general article on the subject, Wintz ⁽¹⁾ describes the commonly used procedure of subdividing the image into subpictures and then computing the transform of the pixels in each subpicture separately. The subpicture size referred to in this paper is 8 x 8 pixels.

The basis for the interest in transform coding is that, for a variety of transforms, picture information is concentrated into fewer transform coefficients than the pixel amplitudes that are transformed. By efficiently assigning bits to the transform coefficients a substantial reduction in the number of bits over those required for conventional pulse code modulation (PCM) of the pixels is obtained. Reductions from 6 bits/pixel PCM to 1-2 bits/pixel (average over the subpicture) in the transform domain is typical.

Of the two-dimensional transforms, the Hadamard transform allows for the fastest implementations in real-time.⁽²⁾ Recent advances in charge coupled device (CCD) technology have made possible the use of the cosine transform at pixel rates in the 1-5 MHz range.⁽³⁾ The differences in the mean square error performance of the various transforms when used to encode images having exponential correlation statistics are in the range of .05% at the .25% error level.⁽⁴⁾ In choosing a transform for image coding the paramount question is one of cost for the given speed requirements.

The Hadamard transform has been implemented in real-time⁽⁵⁾ using TTL hardware in the pipeline FHT configuration. For cases where a slightly lower resolution of 256 elements per scan line or lower frame rates of 7.5 frames per second are acceptable, a combination of two 8 point Hadamard transforms with intermediate random access memory (RAM) may be used. This combination requires only two arithmetic accumulators.

The Hadamard Transform

The following discussion will address the 8 point Hadamard transform. Larger point transforms are common, but for purposes of illustration the 8 point transform is easier to use. Furthermore, there is evidence that there is no substantial improvement in performance for the Hadamard transform for subpicture sizes greater than 8 x 8 pixels.⁽⁴⁾ The 64 point two-dimensional transform may be obtained by cascading two 8 point transforms.

The 8 point Hadamard transform is described by eight equations of the type shown in equation (1), which is the sequency 5 Hadamard coefficient.

$$H_5 = P_0 - P_1 - P_2 + P_3 - P_4 + P_5 + P_6 - P_7 \quad (1)$$

As can be seen, each Hadamard coefficient is generated by a combination of sums and differences of the eight pixel amplitudes. Equation (2) is the matrix operator⁽⁶⁾ that defines the addition and subtraction operations for all eight Hadamard coefficients.

$$[H]_8 = \begin{bmatrix} + & + & + & + & + & + & + & + \\ + & - & + & - & + & - & + & - \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & + & + & + & - & - & - & - \\ + & - & + & - & - & + & - & + \\ + & + & - & - & - & - & + & + \\ + & - & - & + & - & + & + & - \end{bmatrix} \begin{matrix} 0 \\ 7 \\ 3 \\ 4 \\ 1 \\ 6 \\ 2 \\ 5 \end{matrix} \quad (2)$$

The only operations required to implement the Hadamard transform in hardware are addition/subtraction and register shifts (multiplication/division by 2).

The Fast Hadamard Transform

The fast Hadamard transform takes advantage of intermediate results in the computation of coefficients to reduce the total number of computations required. The implementation is very similar to the fast Fourier transform which requires multiplication of complex numbers to implement.

The Pipeline Implementation

Fig. 1 illustrates the pipeline implementation of the FHT. The dashed line input to each arithmetic element designated with a minus sign (-) is the subtrahend input to the indicated subtraction. The figure is organized in the perfect shuffle configuration.⁽⁶⁾ Each stage of the computation is interconnected in the same way. The speed of this method is limited only by the speed of the arithmetic hardware used. Each symbol in Fig. 1 represents an arithmetic element together with a register. Arithmetic operation speeds on the order of 25 ns are available using emitter coupled logic (ECL) 10,000 series devices. For an 8 point transform this corresponds to a pixel transformation rate greater than 300 MHz. The system implemented at NASA-Ames Research Center⁽⁵⁾ uses a 16 point pipeline Hadamard transform together with a multiplexed 4 point implementation to achieve a 128 megapixel per second transformation rate. The system uses TTL hardware operating at 125 ns for each arithmetic operation. Such speeds are required in special applications where it is not possible to process pixels at a steady rate. That such high speeds are not required may be made apparent by considering a 525 line 30 frame per second video signal that has been digitized to 512 pixels per line. The pixel rate is $512 \times 525 \times 30 \approx 8$ MHz. Clearly, when it is feasible to buffer the video data properly, the number of arithmetic elements used may be traded against lower speed requirements.

The Iterative Implementation of the Perfect Shuffle

Since the interconnection pattern of the perfect shuffle is the same for all stages of the computation, the hardware can be reduced to one stage by feeding the output back to the input and cycling the operation 3 times. In general, the number of cycle times required is $\log_2 N$, where N is the number of points to be transformed. This cycling operation is illustrated in Fig. 2 with a feedback loop to show that the output of each arithmetic element is feed back to the corresponding point above the element. Every third operation new data is entered through the multiplexer (MUX). This configuration operates at one third the rate of the pipeline system, but requires only eight arithmetic elements. A pixel rate of 100 MHz can be achieved using ECL 10,000 series components. Using low power, Schottky TTL pixel rates of 20 MHz can be maintained. Based on the calculation in the preceeding paragraph, these speeds are still in excess of the 8 MHz required. The main reason for using such high speed processing has been the difficulty in fully buffering the pixel data. To transform 525 line interlaced video may require a field storage with access to 4 lines of video data in each of two fields. Once the required data is collected it is usually available at the 8 MHz rate from many outputs simultaneously. In the case of a 64 point (8×8) transform a total of 8 inputs at 8 MHz would result since the output of eight scan lines would have to be processed simultaneously. To avoid this requirement, eight lines of pixel data must be stored and read out one subpicture at a time. For 512 pixels by 8 lines this amounts to 4K

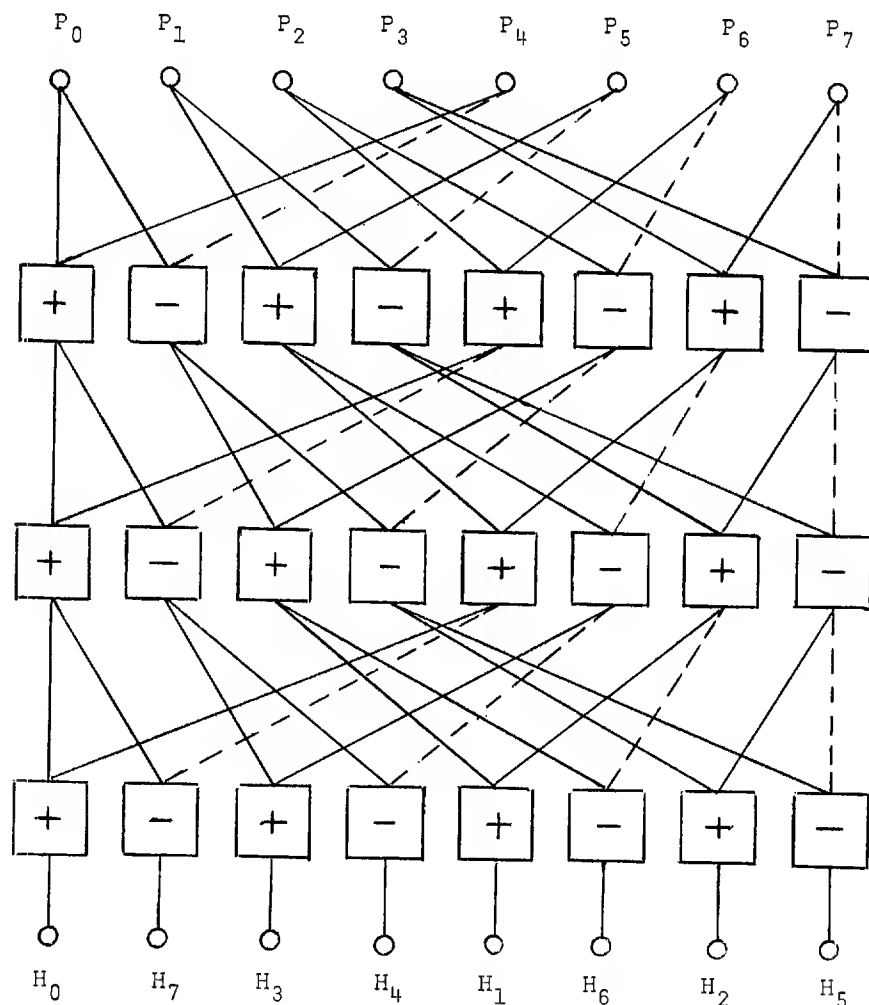


Fig. 1 Pipeline Implementation of the FHT

storage with a read-modify-write speed of 125 ns to meet the requirements for 30 frame per second video. At 6 bits/pixel this amounts to 24 1K TTL RAM chips. In addition to the chip count, these high speed RAMS will consume 15-20 watts of DC power.

An 8 Point FHT Using One Arithmetic Element

Except for the high speed memory requirement a single ECL arithmetic element could perform the required $8 \log_2 8 = 24$ additions and subtractions fast enough to transform pixels at the required 8 MHz rate. At 25 ns for an addition operation the 24 operations could be completed in 600 ns. This equates to 75 ns per pixel or a 13.3 MHz rate, which is more than adequate. Several very high speed ECL memories are required to manipulate the data to be operated on by the ECL arithmetic element. A possible sequence for that manipulation was outlined by Pratt.⁽²⁾

An Economical Implementation for Low Frame Rates

The preceding section discussed several implementations of the FHT that operate at pixel rates in excess of 10 MHz. If the application is suitable for low frame rates (7.5 frames per second or less) or lower resolution (256 x 256 pixels per frame), then a simple implementation of a 64 point (8 x 8) transform can be achieved in real-time.

Fig. 3 illustrates the use of two 8 point transforms using $N^2 = 64$ computations each and a RAM to process the video data.

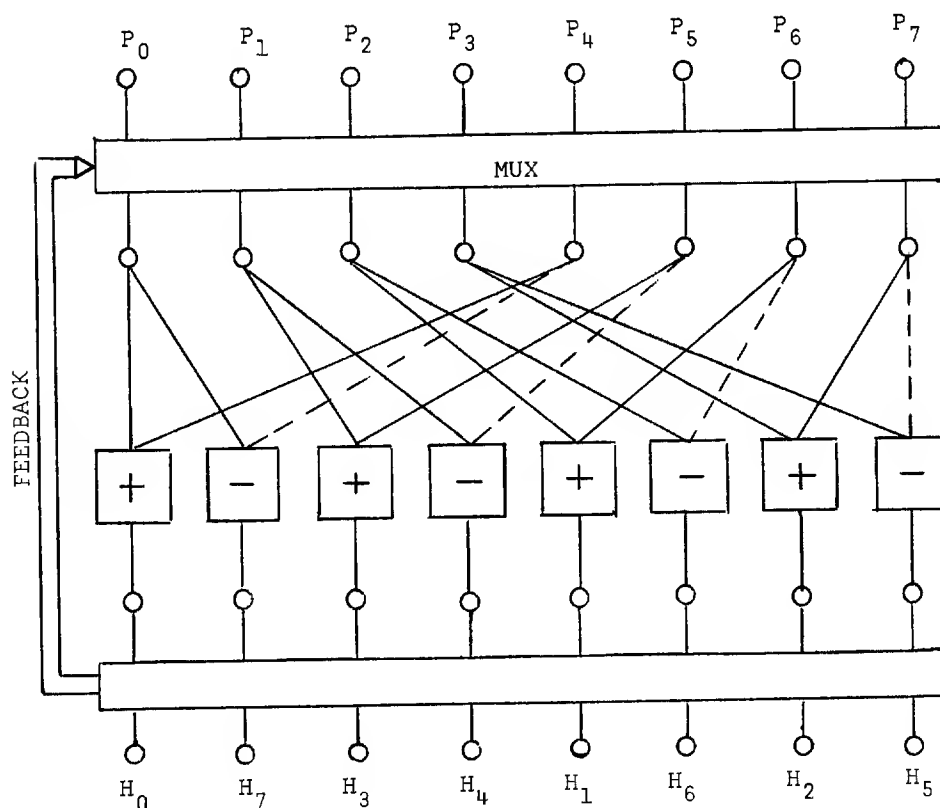


Fig. 2 An Iterative Implementation of the FHT

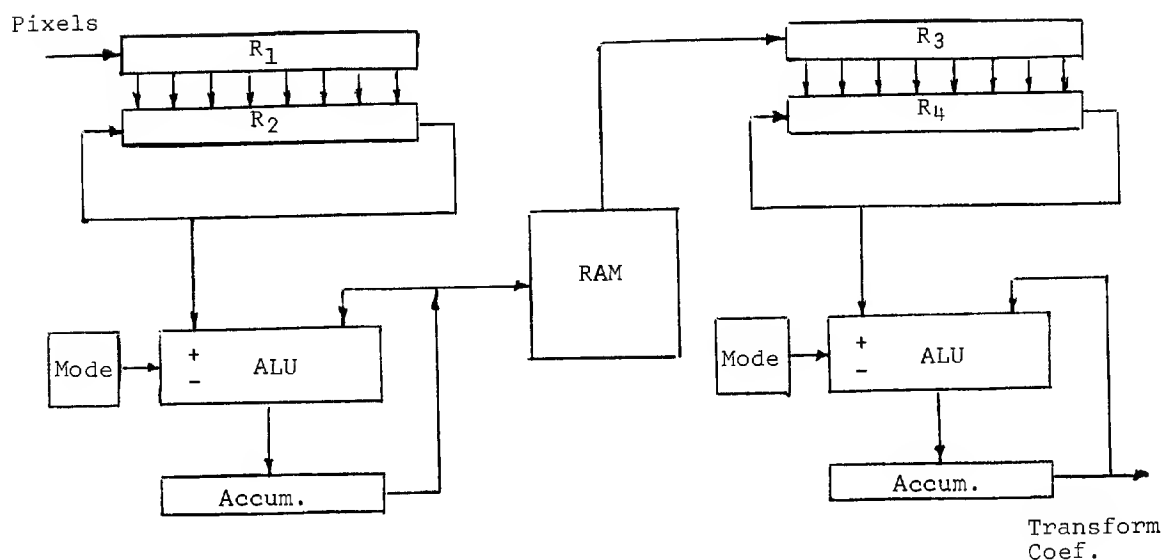


Fig. 3 A Two-Dimensional Implementation of the Hadamard Transform Using Two ALU's and a RAM

Circuit Operation

Video data in the form of pixels enters R_1 . After the accumulation of 8 pixels they are loaded into R_2 and recirculated in R_2 8 times at a rate 8 times faster than the shift rate in R_1 . During each circulation of the 8 pixels the mode input to the arithmetic logic unit (ALU) provides the proper sequence of add (+) and subtract (-) instructions to the ALU to generate a Hadamard coefficient. At the end of each circulation the Hadamard coefficient is transferred from the accumulator into the RAM. The RAM is large enough to store 8 lines of transformed video data. This amounts to 4K words for 512 elements per line. The data is read from the RAM orthogonal to the orientation in which the data was stored. This procedure generates the two-dimensional Hadamard transform coefficients when the data from the RAM is processed through the second 8 point transform circuit.

Speed of Operation

The primary limitation in the speed of operation of this circuit is the RAM. A read-modify-write time of 500 ns will result in a 7.5 frame per second processing rate. Schottky TTL or ECL logic will meet the 62 ns operating speed required to implement the 8 point transform at this speed (8 operations \times 62 ns \approx 500 ns). For this lower speed operation, less than 50 IC's are required to implement this two-dimensional Hadamard transform. The use of ECL to build this unit would result in a pixel rate of greater than 5 MHz. This would be 20 frames per second at a resolution of 500 \times 500 pixels. The use of higher speed RAM's (200 ns read-modify-write) would be necessary to build the system. 4K RAM's are not generally available with this speed. An increase in the number of RAM IC's from 8 to 24 would be required.

Conclusions

Several methods of computing the Hadamard transform have been illustrated covering a wide range of computational speeds. In particular, a combination of two 8 point transformers using the straight forward $N^2 = 64$ computation and a small RAM results in an economical implementation when lower frame rates or resolution are adequate to meet the requirement.

References

1. Wintz, P. A., "Transform Image Coding," Proceedings of the IEEE, Vol. 60, No. 7, pp. 809-820, July 1972.
2. Pratt, W. K., Kane, J., and Andrews, H. C., "Hadamard Transform Image Coding," Proceedings of the IEEE, Vol. 57, No. 1, pp. 58-68, January 1969.
3. Habibi, A., Pratt, W. K., Robinson, G., Means, R., Whitehouse, H., and Speiser, J., "Real Time Image Redundancy Reduction Using Transform Coding Techniques," 1974 International Communications Conference, June 1974.
4. Ahmed, N., Natarajan, T., and Rao, K. R., "Discrete Cosine Transform," IEEE Transactions on Computers, Vol. C-23, pp. 90-93, January 1974.
5. Noble, S. C., Knauer, S. C., and Glem, J. I., "A Real-Time Hadamard Transform System for Spacial and Temporal Redundancy Reduction in Television," presented at 1973 International Telemetering Conference, October 1973.
6. Robinson, G. S., "Orthogonal Transform Feasibility Study," prepared for NASA/MSC by COMSAT, Contract No. NAS 9-11240, Final Report, pp. 3-7 to 3-21, October 1971.

A TEACHING STEREO-VIDEO MICROSCOPE

James F. Butterfield, President
Dimensional Television Corporation
13720 Riverside Drive
Sherman Oaks, California 91423

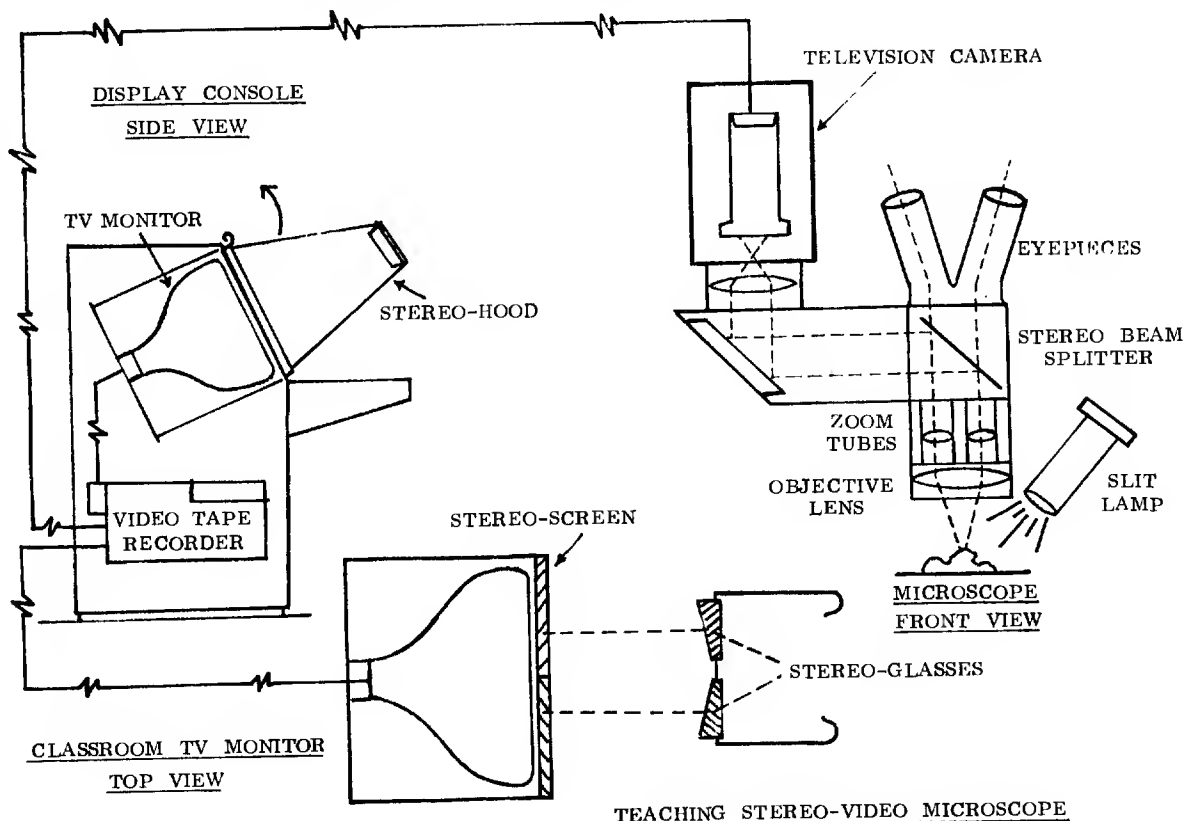
Abstract

A new medical training instrument has been developed, which provides a means whereby microsurgery procedures performed using an optical microscope can be viewed by any number of medical students. This is accomplished by picking-up a television picture from the operating microscope and displaying the picture so the operation can be seen "live" on TV monitors at remote locations. Also the picture can be recorded on video tape for later playback for medical teaching purposes. This instrument has the principal advantage over other teaching means of microsurgery in that it can provide a three dimensional color picture so that the student sees the same view the surgeon saw as he performed the operation.

Background

In recent years an increasing number of surgeons have been using optical stereo microscopes to view small areas of the body so that they can perform microsurgical techniques. Such surgical microscopes are used in the fields of ophthalmology, otolaryngology, neurosurgery and plastic surgery to name a few. However, medical schools and individual surgeons have had difficulty in teaching medical students their techniques and in keeping up with new techniques developed by others. This problem is occasioned by the fact that such surgery is performed in a small area and usually has to be viewed with magnifications of up to 20x. It is, therefore, difficult for a group of students to gather around the patient and attempt to see the same area of the body the surgeon is viewing. The only exception to this is an assistant's microscope trained on the area, which allows one person at a time to see an oblique view.

Surgeons some years ago drew illustrations of surgery and used these as visual teaching aids. More recently still and motion picture cameras have been trained directly on the area of operation or have been attached with beam splitters to operating microscopes. These photographic means have certain disadvantages. Except in the case of Polaroid® photos, the picture is not immediately available as development and printing usually take several days. When the picture is viewed it is sometimes out of focus or some other technical problem occurred, making the photograph or film not useable. Also there is no way for others to see this picture simultaneously with the operation and it must be viewed at a special meeting set up some days later. Finally, these methods provide a picture in only two dimensions and not three dimensional as the surgeon sees with his own two eyes directly or through the stereo optical microscope,



The author and his associates have been active in the fields of video microscopy and three dimensional television (1 thru 6). A recent paper on the development of a surgical video microscope describes an instrument which produces a high resolution stereoscopic color television picture so that the surgeon can actually operate by looking at a television display. This equipment was designed in coordination with and named in honor of Dr. John P. Beale, Jr. of the San Francisco Eye Institute. With such equipment the picture must be three dimensional for the surgeon to perform delicate operations, such as cutting and sewing. The surgical video microscope has the advantage over an optical microscope of permitting the surgeon considerable head movement (7).

Description

The illustration shows the various components of a Teaching Video Microscope. The Slit Lamp directs its light on the patient and the light is in turn reflected up through the Objective Lens and Zoom Tubes of the optical microscope. The eyepiece housing has been separated from the objective lens housing and a Stereo Beam Splitter inserted therein. This transmits half of the light of each image of the stereo-pair up through the Eyepieces and the other half of the light of each image is transmitted horizontally to a Television Camera. The left and right eye images are focused side-by-side on the pickup tubes of the miniaturized color television camera. The two images are displayed side-by-side on a color Television Monitor in a Display Console for "real time" viewing and on classroom TV monitors.

The classroom TV monitors have a Stereo-Screen positioned in front of them. This consists on each half of polarizing filters oriented at right angles to each other. The students wear Stereo-Glasses, which include corresponding polarizing filters in front of each eye that insure the proper image is channeled to the appropriate eye. The Stereo-Glasses also include prism wedges, which cause the images to appear superimposed. Attached to the Display Console monitor is a hinged Stereo-Hood, which consists of an enclosure holding a large pair of Stereo-Glasses. In this case, only individual viewing is possible but the viewer can wear regular corrective glasses and turn away and make notes and look back at the image without the necessity of removing the special Stereo-Glasses. The Display Console also contains a conventional Video Tape Recorder, which is connected into the circuit. This VTR records the two side-by-side pictures and plays them back for stereo viewing on the classroom monitors.

Advantages

The Teaching Video Microscope has several valuable advantages. This is the only means whereby a group of students can see the operation in exactly the same manner as the surgeon does in 3D and color. This viewing can take place simultaneously with the operation or a video tape may be made for later instruction or analysis purposes. The video tape can be edited into a quality educational tape, which may be distributed through medical television network channels.

As the operation is being performed other members of the surgical team can follow the progress of the operation on two dimensional or three dimensional television monitors and thereby be more proficient in aiding the surgeon. Also, other surgeons can follow the progress of the operation from remote areas without having to scrub and suit up. They see the same picture that the surgeon does and hear his comments. Since the tapes are conventionally recorded and played back, this permits easy distribution for medical training purposes in teaching hospitals, research organizations, or for viewing by the doctor in his office or home. The equipment required is that of a conventional video tape recorder and monitor. If the viewing is to be in 3D then a Stereo-Screen and Stereo-Glasses or a Stereo-Hood is necessary. The Hood can be removed or swung up out of the way during two dimensional viewing.

Specialized Teaching Video Microscopes can use "false-color" coding, wherein a black-and-white television picture is picked up and electronically processed so that the different levels of gray provide many shades of color on a color TV monitor. The "false-color" picture is similar to microscopic "staining" and conveys more visual information to the eye than a black-and-white and gray picture. Another possibility is to use electronic polarity reversing so that the picture is seen in its normal shades of black-and-white and then at the touch of a switch the blacks are reversed to white. Sometimes rapid polarity reversing by the surgeon brings out features not seen in a normal manner. Television pickup tubes are now available, which are sensitive to a wide range of the electromagnetic spectrum. For example, there are tubes sensitive to infrared or ultraviolet and these may be used so that heretofore unobservable features can be seen by the surgeon and student. Finally, low light level pickup tubes can be employed for conditions where the intense illumination of a slit lamp would be damaging to the tissues.

Microscopes are following the evolution that has occurred in other fields in that optical instruments, such as motion pictures, are being supplemented and replaced by electronic instruments, such as television. The electronic picture of the Teaching Video Microscope can be stored in a computer, processed and displayed. Electronic enhancement can bring out features not discernable optically.

Conclusion

In the past the surgeon has been extremely limited in his ability to visually communicate information regarding operational techniques on small areas of the body. Now television has revolutionized this by converting the optical picture into an electronic picture, which can be sent down a cable, through the air, and recorded on magnetic tape to be seen now or at a later time. The Teaching Video Microscope can provide a three dimensional color television display of surgical procedures so that the student sees essentially the same picture that the surgeon saw during the time he performed the operation.

References

1. Butterfield, J., "Video Microscope", S. P. I. E. Seminar on Imaging Techniques for Testing and Inspection, February, 1972.
2. Butterfield, J., "Television Microscopy", Preprints of 25th Annual S. P. S. E. Conference, p. 1-4, May, 1972.
3. Butterfield, J., "Three Dimensional Television", Proc. of 15th Annual S. P. I. E. Symposium, Vol. 3, p. 3-9, September, 1970.
4. Butterfield, J., "A Survey of Three Dimensional Displays", Digest of IEEE International Convention, p. 116-117, March, 1972.
5. Butterfield, J., "Stereoscopic Television", Digest of OSA Tech. Papers of "Design & Visual Interface of Biocular Systems", p. ThA7-1, May, 1972.
6. Butterfield, J., "Stereo Television Microscope", U. S. Patent No. 3,818,125, June, 1974.
7. Butterfield, J., "Development of a Surgical Stereo-Video Microscope", Proc. of 18th Annual S. P. I. E. Tech. Meeting, Vol. 54, p. 8-11, August, 1974.

PICTORIAL INFORMATION TRANSMISSION THROUGH SIMULATION

Robert T. P. Wang
Honeywell
Marine Systems Division California Center
West Covina, California 91790

Abstract

In an age when man is inundated with information, his natural ability to selectively assimilate the data presented to him has become an indispensable tool for survival. By the same token, man's visual perception limitations have been used to reduce the amount of data needed to reproduce pictorial information designed for his consumption.

A comparative study of two divergent approaches to the problem of providing an optimum amount of video information to a human viewer is discussed. Video communication systems exemplify one area where the approach consists of reducing the data from real world scenes by video data compression algorithms. The opposite approach is found in visual simulators where scenes are constructed synthetically to approach real world realism by adding cues to the basic structure of the digital image representation used. Such simulators are used in groundbased trainers designed to reduce the cost of training operators of expensive equipment. In both situations there is a need to provide realistic video to a human observer. In the quest for optimum pictorial information transmission, simulated scenes are shown to provide some rather unusual, hitherto unexplored, insights and alternatives.

1. 0 Introduction

Communicating through visual imagery has received considerable attention in two rather diverse fields of application -- the efficient transmission of video information over digital communication channels, and the means of providing the visual stimuli needs of groundbased operator trainers. In both cases, a realistic real-time image reconstructed from a minimum number of information bits is desired.

Video data compression begins with an analog image that is sampled, and digitally coded. The resulting bits are then acted upon by algorithms that use a priori knowledge of the local structure of the pictures in time and space. Some processing algorithms also use visual perception properties of the human eye. The result of such processing is a minimal set of data that can be used by its matching receiver to produce a subjectively pleasing picture to the viewer. In this way the required channel bandwidth can be reduced.

Visual imagery displays in simulation trainers are used to provide a sense of realism to the trainees. The simulator must respond to physical inputs just as the operational equipment would with sufficient detail and resolution to supply all the necessary visual cues to the student. Since the imagery is simulated, the "world" which forms the "source" of the imagery must be stored within the simulator as a data base. Using the stored data and the geometry of the scenario, the proper projection of the light rays is computed to provide the desired image. To produce a real-time system, a study to determine the minimum information needed to provide realistic, real-time visual imagery must be undertaken. Therefore, the approach used to simulate visual imagery for a trainer is discussed. Beginning with basic line drawings, factors that add realism to the simulated image are progressively introduced until an image is obtained.

The approaches used to provide a "realistic" or visually pleasing image for digital video data communication systems appears to be diametrically opposed to the approach used in simulation trainers. However, since the goals of both endeavours are ultimately the same, it is important to both fields that the approaches used by one another be understood. Such a cross fertilization of ideas will result in better techniques for both.

The fundamentals of data compression and simulation trainers are outlined in the next two sections followed by a summary of salient principles used. By applying the underlying principles from one field to the other, it is shown that new directions emerge in both endeavors that will potentially aid the advancement of each. A discussion of these methods is given in Section Four.

2. 0 Fundamentals Of Video Data Compression

The generic open-loop video data communication system is shown in Figure 2-1.

The basic nature of the source signal and the perception properties of the human eye play key roles in the development of any video data compression system. For completeness, the underlying nature of both the source and the viewer are reviewed in the following two sub-sections.

2. 1 The Source

The data source is derived from scanning a projection of the visible world onto a two-dimensional

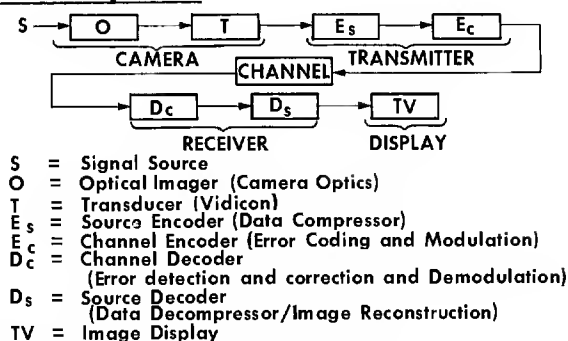


Fig. 2-1. Video Communication System

plane*. The projected data display spatial and temporal properties that can be exploited to reduce signalling rates.

Estournet [1969] showed the video source to be a non-stationary stochastic process. Clustering of iso-intensity picture elements (pixels), forming a source parameter that obey certain area distributions, was studied by Nishikawa, et al [1965]. Both of these properties are inherent in pictures and were used in the development of a generalized structural model of the source [Wang, 1972]. In addition to these characteristics, it is known that intensity variations form patterns in the spatial domain that cover non-trivial areas. Thus, correlation between spatially distinct pixels can persist even over large spatial distances [Wang, 1975].

Time sampling of the video source results in a series of frames. ** Within the time span between two successive frames, the amount of motion that produces changes between the two frames is minimal. This results in considerable temporal redundancy.†

2.2 The Viewer

The human visual system can be grossly partitioned into four major components. Consider Figure 2-2. The optical system images the "real world" onto a screen called the retina. Transducers (rods and cones) on the retina sample and convert the image into electrical impulses that are transmitted along neurons to the brain. It is at the transducer juncture that man's knowledge becomes extremely dim. Considerable processing is thought to occur between the rods and cones and the optic nerve. This is based on the fact that the number of individual fibres in the optic nerve accounts for only 1% of the total number of sensor elements in the retina [Merchant, 1965]. It is further conjectured that processing might even occur along the optic nerve itself.

Rods and cones are unevenly distributed on the retina. A high density of sensors concentrate around the fovea and drop off rapidly on either side as shown in Figure 2-3. This sampling configuration results in spatial band-limitedness that changes with the distance an object lies off the optical axis of the eye.

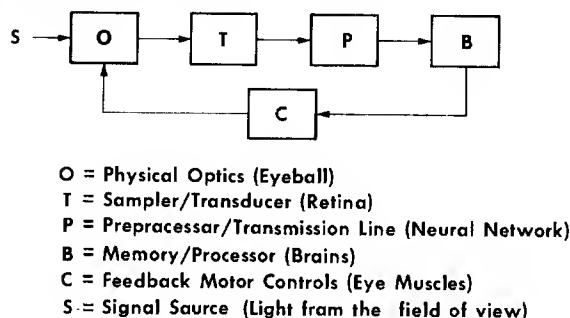


Fig. 2-2. Simplified Block Diagram of the Human Visual System.

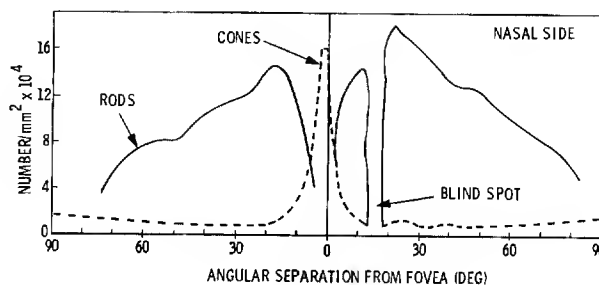


Fig. 2-3. Distribution of Rods and Cones on the Retina.

The human eye is very sensitive to sudden changes that are connected in space to form edges. However, a pattern of edges that form a spatial structure will produce an averaging that makes the eye relatively insensitive to noise having a matching pattern. These phenomena have been modelled by a cross-coupling neural net that produces lateral inhibition [Cornsweet, 1971]. This model also explains the high sensitivity of the eye to motion, noise in low activity regions of a picture and, an insensitivity to noise along edges in a picture.

In the temporal domain, bandlimitedness is manifested through the well known phenomenon of "persistence of vision". Persistence of vision has been the basis for motion in movies and television for many years, allowing movement to be captured in time samples of the pictorial data.

Finally, the eye is responsive to a tri-stimulus color system and becomes progressively color blind as the spatial size of a colored object diminishes. This fact was used in the design of the NTSC standard where high spatial frequencies are represented by the luminance signal alone. A further experimental fact concerning colored imagery is that the eye is more tolerant of luminance noise and error when color is present.

The factors discussed in the previous two sections will be used to support the reasons underlying the various video data compression techniques that have been developed through the years.

2.3 Basic Compression Approaches

2.3.1 Overview. Data compression algorithms initially concentrated on the characteristics of pictures, but later expanded to include the properties of human visual perception.

* The visible world is assumed to include imagery that is not necessarily visible to the naked eye, but which can be made visible via various processing means.

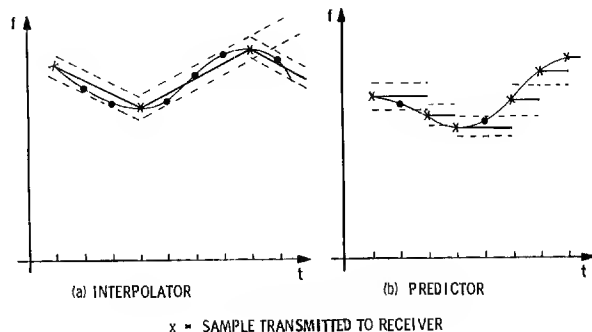
** American NTSC standard frame repetition rate is 30 frames per second.

† It has been shown that camera pan and rapid scene changes do occur, and that frame-to-frame redundancies are not as extensive as earlier expected.

Predominant among early techniques were those that used iso-intensity pixel clustering or "redundancy" in the spatial domain*. As memories or storage devices became economically reasonable, frame-to-frame redundancies were explored to provide further savings.

Context-dependent noise sensitivity of human visual perception was first explored through non-linear quantization of difference signals. Such perception-matched noise distribution was finally carried into the two-dimensional transform-domain. Algorithms tailored to visual perception properties were later extended into the temporal domain through motion dependent spatial coding. A controversy still rages as to whether the "inability" of the eye to perceive high spatial structure in moving objects is truly a characteristic of visual perception. Some researchers feel that the phenomenon, presently adopted as a visual perception trait, is actually induced by long hours of viewing television, whose cathode ray tube introduces lags due to phosphor and electron beam characteristics.

2.3.2 Redundancy Removal Techniques. Given that X_1, X_2, \dots, X_n denotes a string of n pixel values sampled from a picture through a raster scan, the two basic options available to the designer are depicted in Figure 2-4.



2.3.2.1 Interpolator. An interpolator forms a function $f(t:m, n)$ through two preselected pixels X_m, X_n , where X_m is some pixel defining the initial value of the function and X_n the latest value of that function. A term-by-term error analysis is then conducted, where

$$X_t - f_m(t:m, n) = e_t, \quad m \leq t < n \quad (2.1)$$

is evaluated and tested against an error bound e_b . If any $e_t > e_b$, then the values X_{n-1} and X_n are both transmitted to the receiver as the terminal value of the previous interpolation interval and the start of a new interpolation interval respectively. The process is then repeated for the next series of pixels that obey function $f_{n-1}(t:m, n)$, where the subscript denotes the starting location of the function. In this way, the picture is described by a series of functions $\{f_m, f_{n-1}, \dots\}$.

2.3.2.2 Predictor. A similar breaking-up of the picture into a series of functions is accomplished by prediction. In this algorithm, a function of the previous

Fig. 2-4. Redundancy Removal Algorithms.

s pixel values are used to predict the value of the latest sample. Mathematically,

$$\hat{X}_n = f(X_{n-1}, \dots, X_{n-s}) \quad (2.2)$$

If e_b is the error bound for the system and

$$|X_n - \hat{X}_n| < e_b \quad (2.3)$$

then the sampled value X_n , or the difference signal $X_n - \hat{X}_n$, is sent to the receiver. At the receiver, the new value X_n , or the difference signal $X_n - \hat{X}_n$ added to \hat{X}_{n-1} , is used as the value of the new pixel in the picture. The difference manifestation of predictive coding has been proposed in many forms, variously known as Delta Modulation and Differential Pulse Code Modulation (DPCM), with each having many variations to the basic theme. By non-linearly quantizing the step size $X_n - \hat{X}_n$, the quantizing noise energy can be distributed in a manner that is innocuous to the eye.

A point worthy of note is that the previous pixels chosen to calculate \hat{X}_n is not restricted in any way, giving rise to successful predictors that use both two- and three-dimensional redundancy features of the video source.

2.3.3 Context Dependent Noise Shaping. Considerable redundancy is removed by converting the set of samples $\{X_n, X_{n-1}, \dots\}$, to a set of difference signals $\{Y_n, Y_{n-1}, \dots\}$, where

$$Y_i = X_i - X_{i-1}, \quad i \leq n \quad (2.4)$$

By using a non-linear quantizer of the type shown in Figure 2-5, large quantization errors are relegated to edge regions while small errors occur in large regions of slowly varying intensity. This matches the error sensitivities of the eye, giving a subjectively pleasing picture.

Conjecturing that the eye cannot perceive very fine structural detail due to the spatial bandlimitedness of the eye, Landau and Slepian [1971] proposed a transform-domain quantization algorithm that uses this concept.

* For all reference material in this section see the extensive bibliography by William K. Pratt [1973].

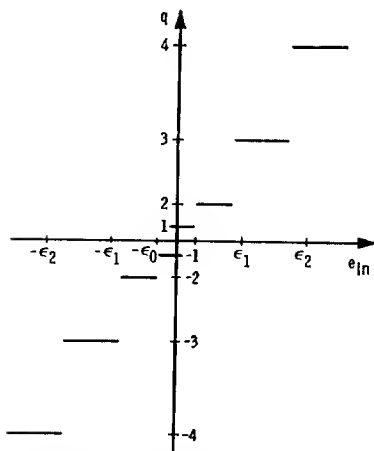


Fig. 2-5. 3-Bit Non-Linear DPCM Quantizer Transfer Function

A picture is divided into a set of 4x4 subpictures, $[f(x, y)]$. The two-dimensional Hadamard transform [Andrews, 1970] of each subpicture is computed

$$[F(x, y, u, v)] = [H(u, v)] [f(x, y)] [H(u, v)] \quad (2.5)$$

By quantizing the Hadamard coefficients in a progressively coarser manner as the sequency of the corresponding vector increases, finer structures reproduce with greater error than coarser structures. Compression is achieved by the net lower number of bits required to represent the higher sequency vector coefficients. It is interesting to note that both the DPCM and the Landau and Slepian algorithm produce good quality pictures at 3-bits per pixel.

Finally, variable resolution algorithms have also been proposed. In these algorithms, regions detected to be in motion are reproduced at subresolution rates while still regions are reproduced with full resolution.

2.4 Summary

The principal features exploited by video data compression algorithms are:

- A. Picture Structure
 1. Pels with like intensities cluster to form finite sized areas - existence of spatial correlation
 2. Spatial redundancies can be represented in terms of a patch work of low-order functions
 3. Frame-to-frame correlation of individual pixels
- B. Visual Perception Properties
 1. Sensitive to noise in low detail areas
 2. Sensitive to noise with structure counter to pictorial structure
 3. Insensitive to random noise or structured noise whose structure matches that of the image
 4. Spatially and temporally bandlimited
 5. Responds to tri-stimulus color
 6. Color sensitivity is image area dependent

3.0 Fundamentals Of Visual Simulation

3.1 The Trainer Scenario

The visual system requirements for an aircraft pilot trainer will be used to illustrate the general constraints imposed by trainers as well as the basic methodology of this field of application.

A typical scenario for an aircraft pilot trainer is shown in Figure 3-1. The operational condition simulated is an aircraft flying over some terrain with Radar and other imaging devices such as TV and FLIR (Forward Looking Infra-red) available to the pilot. In order to provide realistic simulated images on the groundbased trainer's displays, many components of the physical world that affect the image must be modelled.

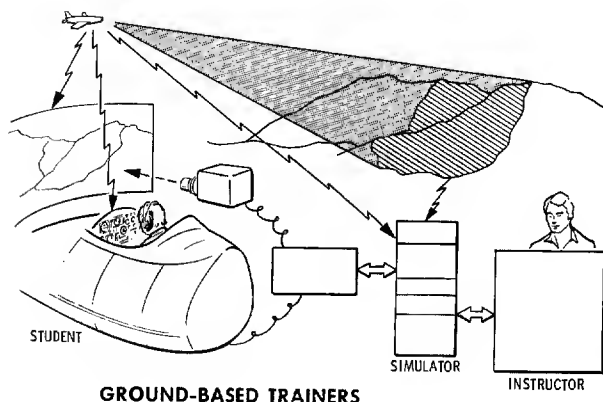


Fig. 3-1. Groundbased Trainers.

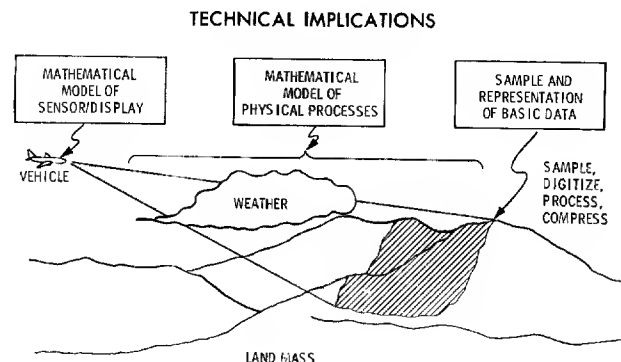


Fig. 3-2. Modelling.

Restricting this discussion to visual simulation, consider Figure 3-2. Data on the terrain flown over must be available and properly referenced with respect to the physical location of the aircraft. This data is typically stored in mass memory to constitute the basic data base. In the "real world", light reflected from the landmass and the objects on it are collected by an optical device to form images. The simulator models the characteristics of the optics and the sensor used in the operational system to include the effects of the equipment on the displayed image. The effects of secondary factors such as weather condition and light sources (e.g., the sun, celestial bodies and artificial lighting on the ground) are also modelled. Finally, since the pilot "flies" the trainer, the aerodynamics of the aircraft must be simulated to realistically respond to the pilot's command. The result of changing the aircraft's altitude and flight conditions must be reflected in the visual image displayed on the monitors and the instruments in the trainer. In many complex trainers, the mock-up cockpit is physically moved to simulate proper physical sensations. Figure 3-3 shows how the major building blocks of a simulation trainer interact with one another. The instructor usually has the ability to control the scenario in various ways and to continually monitor the student's performance. Problem initialization, malfunction and weather condition insertion are but a few of the ways in which the instructor can control training conditions.

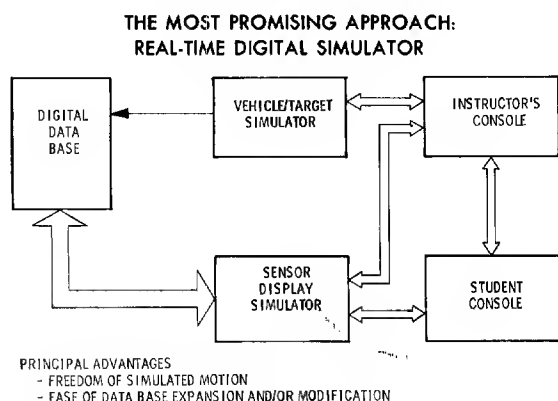


Fig. 3-3. Block Diagram of Trainer Systems

Using the above trainer configuration, it is now possible to discuss the constraints flight trainers place on visual image simulation.

3.2 Trainer System Constraints

As a basis for comparison, consider the side-by-side photographs in Figure 3-4 of a Radar display over the Seattle, Washington area at two different aircraft altitudes. Figure 3-4a shows the radar display at an altitude of 30,000 ft. The three-dimensional nature of the simulation becomes evident when the "shadows" of Figure 3-4a are compared with that in Figure 3-4b which is a simulation of the same aircraft flying over the same geographic location, but at 7,000 ft. In the second display, the STC (Sensitivity Time Control) is adjusted to eliminate blooming.

The simulation system* that produced the pictures in Figure 3-4 has the ability to fly over the complete

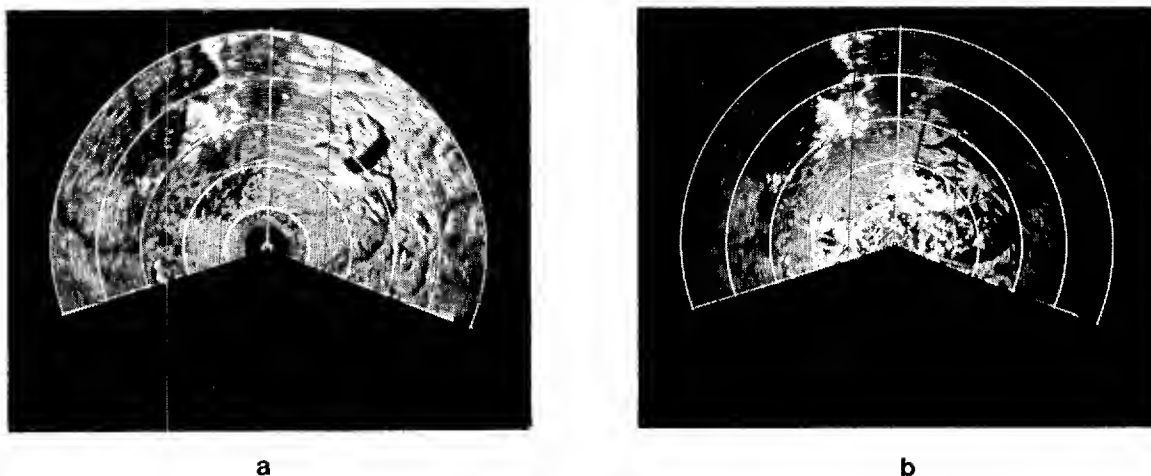


Fig. 3-4. Simulated Radar Over Seattle, Washington.

continental United States at speeds up to Mach 2. The data base which carries elevation data at 250-ft intervals along with reflectance and other data, contains 7×10^9 bits. With the Radar's given pulse repetition rates, a new piece of data is calculated every 900 nanoseconds. It is known that this system* is presently near the limits of the state-of-the-art. It is known that to provide visual data for a TV system, new data must be available at 125 nanosecond-intervals. Furthermore, given a zooming capability, resolutions of as low as a few inches will be required at high information content regions.

* The system referred to is the Honeywell developed Undergraduate Navigator Training System (UNTS).

It is obvious that full visual resolution for trainers is not possible with CGI (Computer Generated Imagery) systems today. What then must be provided in the imagery for the student to benefit from the imagery presented? This problem is addressed through a discussion of simulation techniques.

3.3 Introducing Realism to Simulated Visual Imagery

Perspective projections of objects onto a two-dimensional plane forms the underlying process in any image simulation problem. Figure 3-5 depicts the geometries involved in the projection operation.

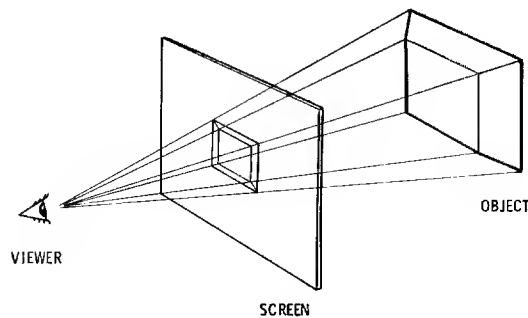


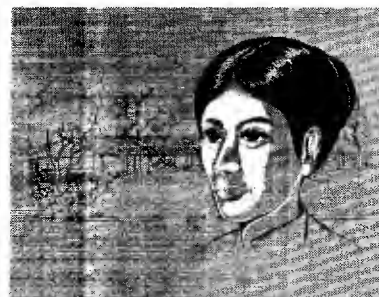
Fig. 3-5. Perspective Geometry

The simplest object representation consists of straightline sections that build-up the outline of an object. By solving the geometry of the problem, a transparent stick figure is projected onto the image plane. Assuming that all regions enclosed by a set of connected straight lines are an opaque plane, then algorithms can be constructed to eliminate "hidden lines" from the projection to produce a "solid" figure. Although the set of objects can be described by curved lines, it is computationally expedient to use straightline segments. To describe a curved line, therefore, requires the length of each approximating segment to be progressively reduced until it is beyond the resolving ability of the eye to perceive the elementary segments.

Consider the picture in Figure 3-6. Note that as one progresses from the line drawing of Figure 3-6a to the final version of Figure 3-6d, the picture becomes progressively realistic, or "life-like". The first step of this process is to add flat grey coloring to introduce body. Then, by progressively adding finer shading, curved surfaces and general lighting effects can be



a



b



c



d

Fig. 3-6.

simulated. By adding highlights, the effect of reflections from point light sources such as the sun and artificial lighting is introduced. Finally, by breaking up the straight edges, the picture becomes less stylized. Concurrent with these steps is the addition of background structure to produce the effect of low resolution data caused by relative distances.

Although it is possible to introduce all of the embellishments just discussed to produce a realistic picture, the required processing is time consuming. Therefore, new techniques are needed to produce more realistic simulated imagery in real time.

3.3 Summary

Generating simulated imagery usually follows this pattern:

1. Data base contains line-segment descriptions of the objects in the "world"
2. Objects in the data base are projected onto a two-dimensional plane in perspective subject to the geometry of the scenario
3. Geometry calculations are performed to
 - a. Eliminate hidden lines (and objects)
 - b. Determine shadowing, shading and highlighting
4. Introduce shadowing, shading and highlighting in accordance to the calculations in (3.)
5. Introduce noise to soften edges and grey level quantization noise.

A final note is that color information is usually contained in the data base. All color calculations can then be carried from the beginning.

4.0 Video Data Compression Via Simulation

4.1 Comparing Video Data Compression and Simulation Techniques

To date, data compression algorithms have operated on the premise that all operations must be based completely on the input waveform of the video signal without investigating the causes of the waveform received. Simulators, on the other hand, begin with a knowledge of the layout of the "world" to be simulated, and then proper signals are generated to produce the desired image on the display. In both instances the ultimate user is the human viewer, so the resulting imagery need only satisfy the subjective requirements of the human eye.

A tabulation of signal processing techniques used in the two fields of development under discussion is given in Table 4-1. The first six functions described are used to produce more subjectively pleasing pictures. Although some techniques can be used by both applications, they are not due to differences in the basic approach. In particular, there are presently no data compression techniques that extract the geometry of the scenario to predict such effects as shadowing, high-lighting and general structure of detail in a picture. By the same token, well known techniques for shaping quantization noise to contextual information to produce a subjectively pleasing picture has not been investigated in simulated imagery studies. As a start on the road to the cross-fertilization of the two areas of study, the need for a new data compression algorithm that uses some visual simulation techniques is proposed to stimulate thinking in this direction.

Function	Table 4-1.			
	Data Compression		Simulation	
	Used	Example	Used	Example
<ul style="list-style-type: none"> • Randomizing • Error shaping <ul style="list-style-type: none"> a) Edge b) Structure • Shading • Highlighting • Edge emphasis • Context dependent resolution • Synthetic structure • Predicting structure effects 	✓	Dithering	✓	Addition of noise
	✓	DPCM		
	✓	Transform coding	✓	Linear fill-in
			✓	Problem geometry
	✓	High pass filtering		
	✓	Transform coding, motion coding	✓	Pattern fill-in
			✓	Problem scenario/geometric calculations
			✓	Problem scenario
<ul style="list-style-type: none"> • Memory or past history effects • Frame-to-frame processing • Correlation 	✓	Frame-to-frame encoding		
	✓	Predictors		

4.2 A Data Compression Algorithm Using Simulation Techniques

The most important element in simulation systems that can be transferred to video data compression systems is the idea of using the geometry of the scenario to achieve video data compression. This idea will be used to explore the following new data compression algorithm.

Among the measurable variables in a given scenario are the locations of the illuminators and the camera, and the magnification (zoom) used in the imaging optics. Define a coordinate system whose origin is at the camera location and one of the axes coinciding with the principal optical axis of the camera. Such a coordinate system converts all linear distances on the image screen to an angular subtend as shown in Figure 4-1.

From the geometry of the problem, the measured subtends and size of shadows and objects in the picture, it is possible to build an estimate of the three-dimensional characteristics of the objects in the field-of-view. Once such an estimate is obtained, subsequent frames of a sequence of pictures generated by a given scene can be predicted through simulation calculations. When the predicted values differ from the projected image in the camera, the

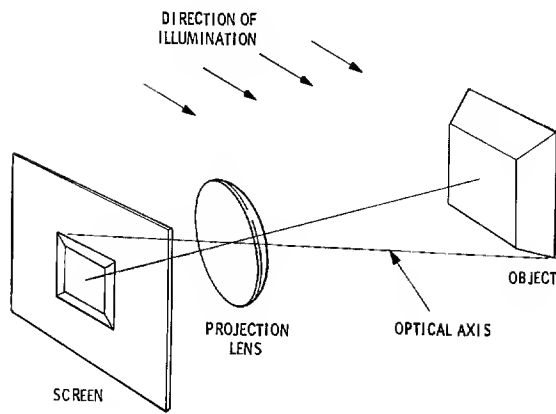


Fig. 4-1.

difference is transmitted to the receiver and the data-base adjusted to correct for error. Changes in the direction and location of the camera then forms an important part of the data that must be sent to the receiver. When long series of pictures are to be taken such as in ERTS (Earth Resources Technology Satellite) and deep space applications, the use of simulation techniques becomes very attractive. The method amounts to a series of processes that begin with pattern recognition, followed by geometric modelling and then finally predictive coding. The algorithms is presently under study and the details will be discussed in subsequent papers. The method is discussed to illustrate one attempt at bridging the two fields. By using one's imagination, it is easy to see how other techniques from one field might be used to enrich the state-of-the-art of the other.

5. 0 Conclusions

Techniques used to provide subjectively pleasing visual imagery for video data compression algorithms and visual simulation for groundbased trainers were discussed. By comparing the methods used in these two fields, it is evident that under certain conditions, some of the methods used in one can be transferred to the other.

Although the example chosen to illustrate the cross-fertilization possibilities addresses the video data compression problem, it is quite evident from Table 4-1 that the reverse operation would be fruitful too. In fact, a study of Table 4-1 reveals many areas that could be explored for new algorithms.

References

- Andrews, H. C. , "Computer Techniques in Image Processing, " Academic Press, N. Y. , 1970.
- Cornsweet, T. N. , "Visual Perception, " Academic Press, N. Y. , 1970.
- Estournet, D. , "Etude Statistique d'un Signal d'image" and "Compression d'Information de Signaux d'Images par les Systemes Differentiels Codes", L'Onde Electrique, 49, fasc. 8, 1969.
- Geokezas, M. ; Wang, R. T. P. , "Bandwidth Reduction Via a Source Selective Variable Resolution Data Compression Algorithm, " presented at the 1975 IEEE International Conference on Communications (ICC '75) June 16-18, 1975.
- Merchant, J. , "Sampling Theory for the Human Visual Senses, " Journal of the Optical Society of America, 55:10, Oct. , 1965.
- Pratt, W. K. , "Bibliography on Digital Image Processing and Related Topics" USCEE Report 453, Electronic Sciences Laboratory, University of So. California, Los Angeles, California, Sept. 1, 1973.
- Landau, H. F. ; Slepian, D. , "Some Computer Experiments in Picture Processing for Bandwidth Reduction, " Bell System Technical Journal 50:5, May-June 1975.
- Nishikawa, S. ; Massa, R. J. ; Mott-Smith, J. C. , "Area Properties of Television Pictures", IEEE Trans. on Information Theory, IT-11, No. 3, 1965.
- Wang, R. T. P. , "A Study of the Structure of Television Pictures as Applied to Practical Data Compression Techniques, " PhD Thesis, Stanford University, California. Also published as Scientific Report No. 41, Center for Radar Astronomy, Stanford Electronics Laboratories, Stanford University, Stanford, California, 1972.
- Wang, R. T. P. , "Autocorrelation Functions of Real Pictures, " In preparation, 1975.

THREE DIMENSIONAL PERCEPTION PRODUCED BY
TWO 2 DIMENSIONAL GEOMETRIC PATTERNS PRESENTED
WITH BRIEF EXPOSURE TIMES AND TIME INTERVALS

Kiyoe Mizusawa
The Pennsylvania State University
Erie, Pennsylvania 16510

Abstract

This paper examines the three dimensional perception of human subjects who observe two 2 dimensional geometric patterns at brief exposure times and time intervals without the aid of a stereoscope. In general, presenting two 2 dimensional geometric patterns at the proper spacing, exposure times, and time intervals produces two dimensional movement perception in human observers. The observer sees a set of patterns moving in a lateral direction, from its first location to its second. The lateral two dimensional movement perception occurs over a wide range of exposure times and time intervals of stimulus patterns. Some geometric patterns presented at certain limited exposure times and time intervals, however, produce three dimensional patterns. The present research investigates the range of the conditions necessary for producing three dimensional pattern recognitions while presenting two dimensional patterns. The research may provide an understanding of human pattern recognition processes and aid in improving man-machine systems.

Introduction

The production of three dimensional as well as two dimensional movement patterns depends upon such physical variables as exposure times of stimuli, X_1 ; time interval between stimuli, X_2 ; intensity of stimuli, X_3 ; and distance between stimuli, X_4 . The present research investigates the optimum conditions for producing the perception of three dimensional movement relative to the perception of two dimensional movement, specifically to find the interrelationships among four variables. The determination of the optimum conditions is made by a least square method.

Experiments

Equipment

The apparatus used to project geometric patterns was a light generator consisting of a combination of an electric pulse generator, electric switches, and light signal sources (1). Stimulus patterns used were a pair of 10 cm lines shown in Figure 1; a set of 3 dots and a set of 4 dots, in Figure 2; a pair of 30 degree open triangles with 10 cm sides, in Fig. 3; a pair of 90 degree open triangles with 10 cm sides, in Figure 4; a pair of half circles with 10 cm diameters, in Figure 5; and a pair of 10 cm diamonds, in Figure 6. These figures were projected on a screen at the eye level of the observer with specially modified projectors. The distance between two stimulus patterns, X , were varied 2, 5, 8, 11 and 15 centimeters. The observation distance was kept at 180 centimeters.

Procedures

Five volunteer subjects from the Introductory Psychology Course served for the experiments. They were paid \$2.50 per hour. Fifteen measurements for each exposure time were taken with each subject. Each subject served five different distance variables, 2, 5, 8, 11 and 15 centimeters.

During the experiment each of the four variables X_1 , X_2 , X_3 and X_4 were varied at increment values. These increment values are designated with the following notation:

$$\begin{aligned} X_1 &= \{X_1^{(1)}, X_1^{(2)}, \dots, X_1^{(J)}, \dots, X_1^{(N_1)}\} \\ X_2 &= \{X_2^{(1)}, X_2^{(2)}, \dots, X_2^{(J)}, \dots, X_2^{(N_2)}\} \\ X_3 &= \{X_3^{(1)}, X_3^{(2)}, \dots, X_3^{(J)}, \dots, X_3^{(N_3)}\} \\ X_4 &= \{X_4^{(1)}, X_4^{(2)}, \dots, X_4^{(J)}, \dots, X_4^{(N_4)}\} \end{aligned} \quad (1)$$

In the above notation, X_1 is set at values of 10, 30, 60, 90, 150 and 210 milliseconds. Then, $X_1^{(1)} = 10$, $X_1^{(2)} = 30$, $X_1^{(3)} = 60$, $X_1^{(4)} = 90$, $X_1^{(5)} = 150$, and $X_1^{(6)} = 210$. The

frequency of responses with which variable X_2 (time interval between stimuli) was chosen by subjects to be "Optimum" in producing perception of three dimensional as well as two dimensional movement was obtained in the experiment while X_2 was at its J^{th} increment value ($X_2 = X_2^{(J)}$) and X_1 , X_3 , and X_4 are at their L^{th} , M^{th} and N^{th} increment values ($X_1 = X_1^{(L)}$, $X_3 = X_3^{(M)}$, and $X_4 = X_4^{(N)}$).

Experiments were conducted in which variables X_1 , X_3 and X_4 remained constant at their first, second, and tenth increment values, respectively. That is, $X_1 = X_1^{(1)}$, $X_3 = X_3^{(2)}$, $X_4 = X_4^{(10)}$ were kept constant. For example, each subject was required to choose the $X_2^{(J)}$, $J = 1, 2, 3, \dots, 10$, which he perceived as the optimum three dimensional and two dimensional stimulus patterns. Each subject was required to make ten consecutive judgments for each condition before being tested with the next stimulus patterns. Response frequencies (f) of five subjects were summarized with respect to the best movement perception in the first series of experiments. Under $X_1^{(2)}$ condition, 11 responses out of 75 responses made by the five subjects were considered "best." Under $X_1^{(3)}$ condition, 45 responses made by the five subjects were considered "best." Under $X_1^{(4)}$ condition, 19 responses made by the five subjects were considered "best." Then the responses were defined as follows:

$$\begin{aligned} f(1,2,10) \left\{ X_1^{(2)} \right\} &= 11 \\ f(1,2,10) \left\{ X_1^{(3)} \right\} &= 45 \\ f(1,2,10) \left\{ X_1^{(4)} \right\} &= 19 \\ \text{and} \quad f(1,2,10) \left\{ X_1^{(J)} \right\} &= 0 \quad \text{for } J = 1, 5, \text{ and } 6. \end{aligned} \quad (2)$$

The notation $f^{(1,2,10)} \left\{ X_1^{(J)} \right\}$ indicates the frequency of the responses judged "best" for each J , $J = 1, 2, 3, \dots, 6$, under the conditions mentioned above.

Sample means of total number of data points were computed by the following definition:

$$\mu^{(L,M,N)} = \frac{\sum_{J=1}^{N_1} X_1^{(J)} f^{(L,M,N)} \left\{ X_1^{(J)} \right\}}{\sum_{J=1}^{N_1} f^{(L,M,N)} \left\{ X_1^{(J)} \right\}} \quad (3)$$

Results and Discussion

The sample means for the optimum two dimensional and three dimensional movement perceptions were summarized in the graph of Figure 7. Each curve in the graph expresses the values of the two dimensional and three dimensional perceptions occurring when the distance between patterns, X_4 , is varied 2, 5, 8, 11 and 15 centimeters. The perceptions of three dimensional movement occurred in the limited ranges which are indicated by dotted square lines in Figure 7.

In Figure 7, the sample means for the optimum level of two dimensional movement perception are expressed by two family curves: the distances between two patterns, 2 cm and 5 cm, indicate polynomial functions and the distances between patterns, 8, 11, and 15 cm, tend to indicate rational functions. In general, when the times between patterns are varied from 70 to 710 milliseconds for all the distance conditions, the optimum two dimensional movement perception occurs when the average exposure time, μ , is somewhere between 20 and 100 milliseconds. The three dimensional movement perception, however, occurs in the range of exposure time, X_1 , somewhere between 3 ms to 82 ms, and time interval, X_2 , somewhere between 4 ms to 200 ms as shown in Figure 7. The detail analyses of the results will appear in a future paper.

In Figures 3, 4, 5, and 6, the left hand side indicates the two 2 dimensional stimulus patterns; the right hand side indicates the perceived patterns of two and three dimensional movement when the distance between two stimulus patterns are varied 2, 5, and 8 centimeters; the top pattern is perceived when the distance between two stimulus patterns

are kept 2 cm; the middle pattern, 5 cm, and the bottom pattern, 8 cm.

Figure 1. Stimulus Patterns

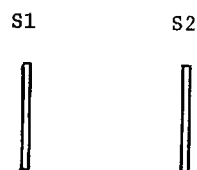
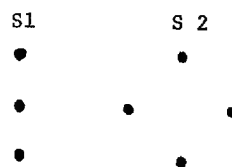


Figure 2



The perception of three dimensional movement reported by subjects exposed to the various stimulus patterns are briefly summarized as follows:

Fig. 1. Two vertical lines

Two lines produce only two dimensional lateral movement perception in all the conditions, and not three dimensional movement.

Fig. 2. A set of three dots and a set of four dots

Three center dots fuse together as a dot that circles around the vertical two dots.

Fig. 3. A pair of two 30 degree open triangles with 10 cm sides

<u>Exposure Times</u>	<u>Time Intervals</u>	<u>Pattern Recognition</u>
16.3 ms	50.0 ms	A dot remains still in the triangles. The open triangles move to and fro.
12.5 ms	48.0 ms	The open triangles fuse together and form a diamond which spins around an axis formed by the open end of the triangle. A dot remains at the pointed corner of the diamond.
11.8 ms	35.0 ms	The diamond makes a fast spinning movement with a fused dot remaining at the corner of the diamond.
10.2 ms	30.6 ms	The diamond makes a fast spinning movement with a fused dot.
5.1 ms	15.3 ms	The diamond remains stationary, but the two dots fuse together as one dot, which moves laterally within the diamond.
4.7 ms	14.1 ms	Two triangles superimpose upon each other with a fused dot at the center.

Fig. 4. A pair of two 90 degree open triangles with 10 cm sides

The perception of three dimensional movement is further intensified by the condition described in Figure 3 under all the conditions.

Fig. 5. A pair of two half circles with 10 cm diameter

The perception of three dimensional movement is smoother than under any of the other conditions described.

Fig. 6. A pair of two 10 cm diamonds

Two diamonds fuse together and move to and fro. A fused dot moves in the opposite direction to the diamond at shorter exposure times. A fused diamond remains stationary, but a fused dot moves in a lateral direction within the diamond.

These are some of the unusual experiences described by the subjects.

Figure 3

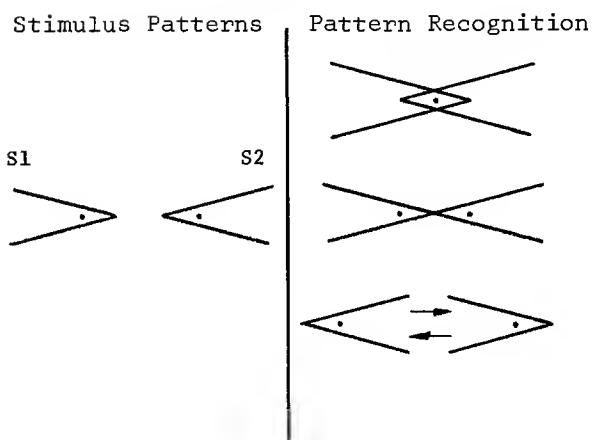


Figure 4

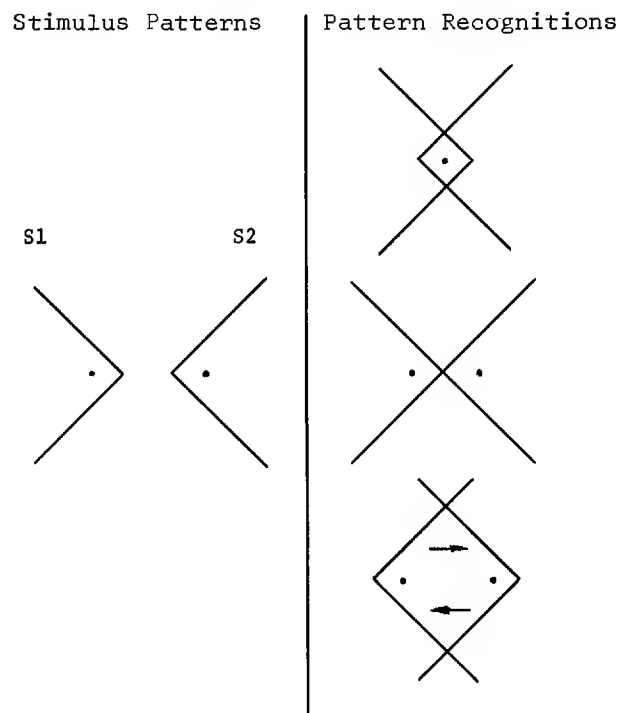


Figure 5

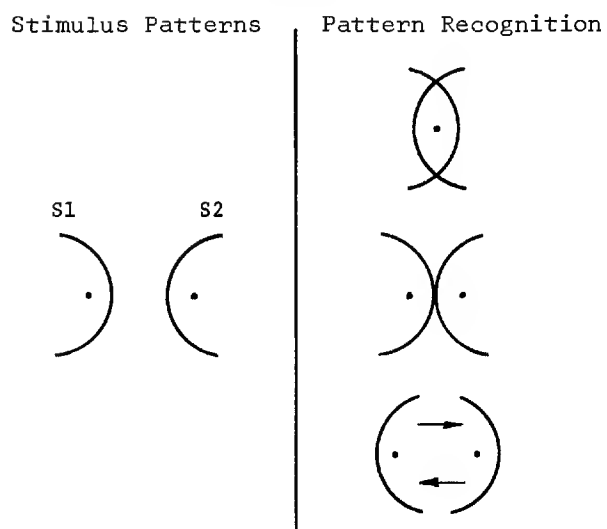
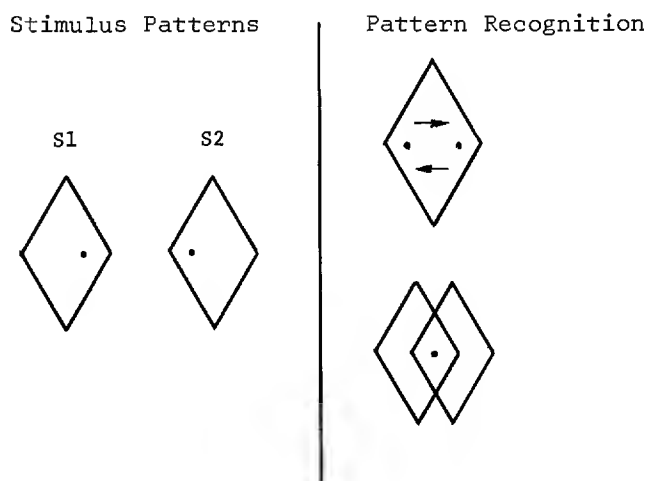


Figure 6



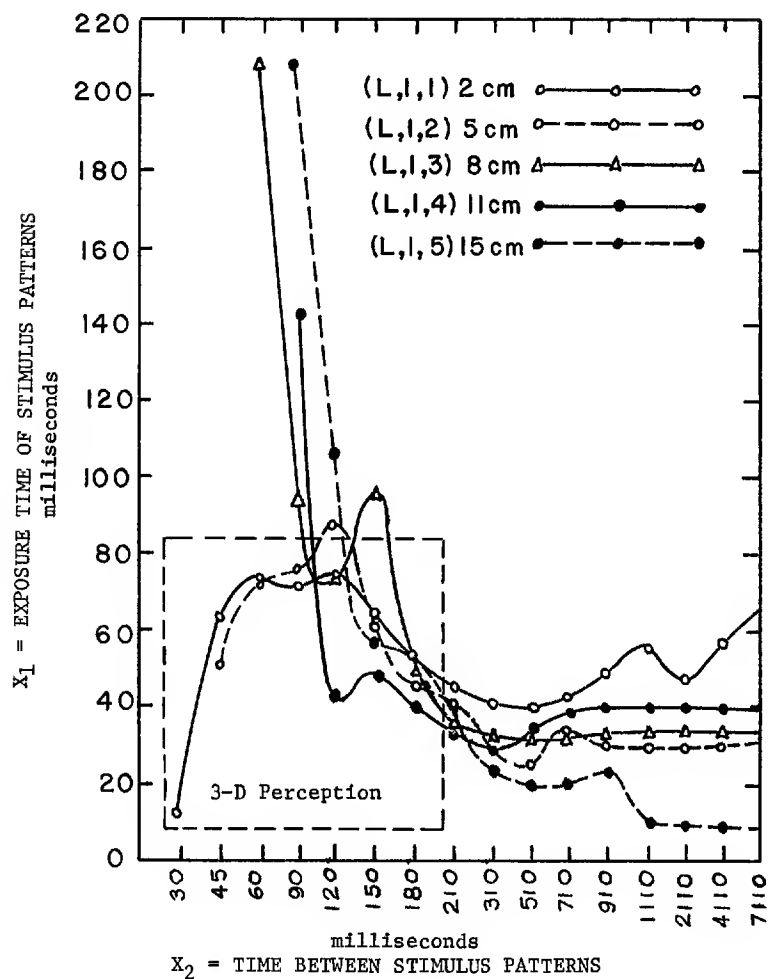


Figure 7. Sample Mean, μ , Expressing the Values of the Optimum Movement Perception in Two Dimension and Three Dimension

The graphs of all the experimental results are not shown here. However, the experimental results lead to the following generalizations about the conditions that create three dimensional movement perception.

1. Distance between stimuli (X_4) increases as intensity of stimuli (X_3) increases; time interval between stimuli (X_2) and exposure time of stimuli (X_1) remaining constant.
2. Intensity of stimuli (X_3) increases as time interval between stimuli (X_2) increases; distance between stimuli (X_4) and exposure time of stimuli (X_1) remaining constant.
3. Distance between stimuli (X_4) increases as time interval between stimuli (X_2) increases; intensity of stimuli (X_3) and exposure time of stimuli (X_1) remaining constant.
4. Exposure time of stimuli (X_1) decreases as time interval of stimuli (X_2) increases; intensity of stimuli (X_3) and distance between stimuli (X_4) remaining constant.

In terms of theory, Gestalt Psychologists might use the theory of cortical irradiation to interpret the results of the experiment, while the photo-chemical physiologist might use the theory of neural interaction at the retinal level to explain the same results. The present experiments do not prove either one of the theories, since the experiments merely demonstrate the three dimensional perception when two 2 dimensional geometric patterns are presented with brief exposure times and time intervals. One of the experiments which might provide some understanding of three dimensional movement perception would relate its occurrence to theories about other visual processes that are well supported by empirical results. For this purpose an experiment is suggested to relate the three dimensional movement perception to the rate of binocular alternation in stereoscopic vision. The correlational study between three dimensional movement perception and the rate of binocular alternation in each individual may provide some interpretation of the present experimental findings.

Conclusion

Human observers do experience three dimensional movement perception when two 2 dimensional geometric patterns are presented with brief exposure times and time intervals. Further studies in the area may aid us to understand one aspect of operator performance and to help in improving man-machine systems. In order to understand theoretical aspects of three dimensional movement perception, a study correlating three dimensional perception to the rate of binocular alternation is also suggested.

References

1. Mizusawa, K., "A Variable Short Impulse Signal Generator," Technical Note, TN-AE-67-166, 1967, Chrysler Corporation Space Division.
2. Mizusawa, K., "Distance Judgment in Space," Journal of the Society of Photo Optical Instrumentation Engineers, Aug.-Sept., 1970.

